НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

Теплоенергетичний факультет

Кафедра автоматизації проектування енергетичних процесів і систем

"На правах рукопису"
УДК _____

«До захисту допущено»
Завідувач кафедри
_____ О.В. Коваль
(підпис) (ініціали, прізвище)
“___”_____2019р.

Магістерська дисертація

зі спеціальності - 122 Комп’ютерні науки
за спеціалізацією - Комп'ютерний моніторинг та геометричне моделювання процесів і систем
на тему: Розробка системи підтримки прийняття рішень при аналізі інформаційного забезпечення безпеки даних енергетичних процесів. Підсистема аналізу

Виконав (-ла): студент (-ка) 6 курсу, групи ТМ-81мп
Лукашевич Анна Євгеніївна _____
(прізвище, ім’я, по батькові) (підпис)

Науковий керівник к.т.н., доцент Ходаковський О. В. _____
(посада, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Рецензент _____ _____
(посада, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.
Студент _____
(підпис)

Київ — 2019

Національний технічний університет України
“Київський політехнічний інститут ім. Ігоря Сікорського”

Факультет теплоенергетичний

Кафедра автоматизації проектування енергетичних процесів і систем

Рівень вищої освіти другий, магістерський

зі спеціальності - 122 Комп’ютерні науки
за спеціалізацією - Комп'ютерний моніторинг та геометричне моделювання процесів і систем

ЗАТВЕРДЖУЮ
Завідувач кафедри
Коваль О.В. _____
(прізвище, ініціали) (підпис)
«____» _____ 2019р.

З А В Д А Н Н Я
НА МАГІСТЕРСЬКУ ДИСЕРТАЦІЮ СТУДЕНТУ

Лукашевич Анна Євгеніївна
(прізвище, ім’я, по батькові)

1. Тема дисертації Розробка системи підтримки прийняття рішень при аналізі інформаційного забезпечення безпеки даних енергетичних процесів та систем. Підсистема аналізу

-

-

Науковий керівник к.т.н., доцент Ходаковський О. В.
(прізвище, ім’я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від “4” листопада 2019 року №3812-с

2. Строк подання студентом дисертації “____” _____201 року_____

3. Об’єкт дослідження система підтримки прийняття рішень при аналізі інформаційного забезпечення безпеки даних

-

-

4. Предмет дослідження система із забезпечення безпеки даних енергетичних процесів

5. Перелік питань, які потрібно розробити

1) проаналізувати сучасні методи забезпечення безпеки даних;
2) проаналізувати існуючі системи із забезпечення безпеки даних;
3) розробити структуру для побудови системи;
4) розробити користувацький інтерфейс;
5) розробити програмне забезпечення.

6. Орієнтований перелік ілюстративного матеріалу актуальність, мета роботи, завдання та методи досліджень, математична задача, структура системи, архітектура програмного продукту, початок роботи, введення вхідних параметрів, результати виконання програми, висновки.
7. Орієнтований перелік публікацій

-

8. Дата видачі завдання «14» січня 2019 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів виконання магістерської дисертації	Строки виконання етапів магістерської дисертації	Примітка
1	Отримання завдання		
2	Збір інформації		
3	Аналіз вимог завдання, вибір методів і засобів розв'язання поставленої задачі		
4	Підготовка матеріалів магістерської роботи		
5	Проміжний контроль підготовки		
6	Підготовка публікацій		
7	Підготовка доповідей на конференціях за темою магістерської роботи		
8	Доповідь на конференції		
9	Написання основних розділів автореферату		
10	Звіт за перший рік роботи над магістерською дисертацією		

Студент _____ Лукашевич А.Є.
(підпис) (прізвище та ініціали)

Науковий керівник _____ Ходаковський О.В.
(підпис) (прізвище та ініціали)

РЕФЕРАТ

Структура й обсяг дипломної роботи. Магістерська дисертація складається зі вступу, п’яти розділів, висновку, переліку посилань з 16 найменуваннь і містить 19 рисунків, 8 таблиць. Повний обсяг магістерської дисертації складає 83 сторінок, з яких перелік посилань займає 2 сторінки.

Актуальність теми. У наш час дуже важливу роль відіграє забезпечення інформаційної безпеки даних. Системи із забезпечення стану захищеності і зберігання даних, при якому забезпечено конфіденційність, доступність і цілісність інформації, або комплекс заходів, спрямованих на забезпечення захищеності інформації від несанкціонованого доступу, використання, оприлюднення, руйнування, внесення змін, ознайомлення, перевірки запису чи знищення, використовуються майже на всіх підприємтсвах та установах. Розвиток комп’ютерних технологій і їх використання в багатьох сферах життя є на сьогодні одним з головних факторів необхідності виникнення систем із забезпечення безпеки. Діяльність із забезпечення інформаційної безпеки здійснюється за допомогою різних способів, засобів і прийомів, які у сукупності й складають методи. Метод передбачає певну послідовність дій на підставі конкретного плану. Методи можуть значно змінюватися і варіюватися в залежності від типу діяльності, в якій вони використовуються, а також сфері застосування.

Мета дослідження полягає в створенні засобів та інструментів для покращенного контролю із забезпечення інформаційної безпеки.

Об’єктом дослідження є здатність системи аналізувати спроби несанкціонованого доступу до інформації та даних та видача користувачу рекомендацій щодо зменшення числа таких спроб або їх уникнення.

Предметом дослідження є комп’ютерні інформаційні технології із забезпечення безпеки даних, аналіз атак та спроби несанкціонованого доступу, аналіз загроз таких атак та видача рекомендацій спеціалісту із безпеки.

Наукова новизна одержаних результатів. Найбільш суттєвими

науковими результатами магістерської дисертації є:

- аналіз загроз та виявлення найбільш небезпечного періоду часу залежно від кількості спроб несанкціонованого доступу;
- видача користувачу рекомендацій щодо уникнення атак на систему відповідно до кількості спроб несанкціонованого доступу.

Практичне значення. Створений програмний продукт, метою якого є аналіз можливих загроз втрати інформації та видача рекомендацій користувачу (спеціалісту із забезпечення безпеки) для їх усунення, забезпечує більш якісний контроль за збереженням цілісності даних та інформації, та може бути використаний у будь-яких напрямках, які потребують відповідного рівня відстеження захищенності інформації.

ABSTRACT

Structure and volume of thesis. The master's dissertation consists of an introduction, 5 sections, a conclusion, a list of references from 16 titles and contains 19 drawings, 8 tables. The full volume of the master's dissertation is 83 pages, of which the list of links takes 2 pages.

Actuality of theme. Nowadays, information security is very important. Security and storage systems that maintain the confidentiality, accessibility and integrity of information, or a set of measures to protect information from unauthorized access, use, disclosure, destruction, modification, familiarization, record verification or destruction, are used almost at all enterprises and institutions. The development of computer technologies and their use in many areas of life is one of the main factors for the need for security systems to emerge. Information security activities are accomplished through a variety of methods, tools and techniques that collectively form the methods. The method involves a certain sequence of actions based on a specific plan. Methods can vary greatly and vary depending on the type of activity in which they are used and the scope. Information security is based on information protection activities − ensuring its confidentiality, accessibility and integrity, as well as preventing any compromise in a critical situation. Such situations include natural, technological and social disasters, computer crashes, physical abduction, and the like.

The purpose of the study is to create tools and applications for improved control over information security.

The object of research is the ability of the system to analyze attempts to gain unauthorized access to information and data and to issue recommendations to the user to reduce or avoid such attempts.

The subject of the research is computer information technology for data security, analysis of attacks and unauthorized access attempts, analysis of threats of such attacks and issuing recommendations to a security specialist.

Scientific novelty of the obtained results. The most significant scientific results of the master's thesis are:

− analysis of threats and detection of the most dangerous period of time depending on the number of attempts of unauthorized access;

− issuing recommendations to the user to avoid system attacks according to the number of attempts of unauthorized access.

Practical meaning. A software product designed to analyze potential threats of information loss and issue recommendations to the user (security specialist) to eliminate them, provide better control over the integrity of data and information, and can be used in any areas that require appropriate level of information security tracking.

ЗМІСТ

Перелік умовних скорочень і позначень ……..9
Вступ 10
1. Постановка задачі..... 13
2. Аналіз проблеми інформаційного забезпечення безпеки даних. Підсистема аналізу 15
2.1. Загальні поняття 15
2.2. Огляд існуючих систем інформаційного забезпечення безпеки даних 17
2.3. Характеристики загроз інформаційної безпеки 23
2.4. Інструмент для забезпечення безпеки даних Firewall 30
2.4.1. Апаратний фаєрвол..... 32
2.4.2. Програмний фаєрвол 33
2.5. Обробка спроб несанкціонованого доступу 34
2.6. Підходи до вирішення задачі аналізу безпеки даних 39
2.7. Розподілена мережева атака типу DDos 41
2.8. Висновки до розділу 2 43
3. Методи реалізації системи підтримки прийняття рішень при аналізі інформаційного забезпечення безпеки даних 44
3.1. Середовище розробки Visual Studio 2019 44
3.2. Інструмент для візуального проектування баз даних MySQL Workbench .. 48
3.3. Система управління базами даних MySQL 53
3.4. Об’єктно-орієнтована мова програмування C# 58
3.5. Висновки до розділу 3 63
4. Взаємодія з користувачем 64
4.1 Опис бази даних системи 64
4.2 Системні вимоги та інсталяція 67
4.3 Сценарій роботи користувача 68
4.4. Висновки до розділу 4 71
5. Стартап проект 72
5.1 Опис ідеї проекту 72

5.2 Технологічний аудит ідеї проекту 74
5.3 Аналіз ринкових можливостей запуску стартап-проекту 75
5.4 Висновки до розділу 5 80
Висновки 81
Список використаних джерел..... 82

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ І ПОЗНАЧЕНЬ

IDE – Integrated Development Environment, Інтегроване середовище розробки

Фреймворк – інфраструктура програмних рішень, що полегшує розробку складних систем

БД – бази даних

СКБД – система керування базами даних

C# – об'єктно-орієнтована мова програмування

ОС – операційна система

DDoS – відмова в обслуговуванні, Distributed Denial of Service

Брандмауер – міжмережевий екран

SIEM – Security information and event management

АСУ – Автоматизована система управління

DLP – Data Leak Prevention, запобігання витікам інформації

ВСТУП

Питання інформаційної безпеки займають особливе місце і в зв'язку зі зростаючою роллю інформації в житті суспільства і вимагають особливої уваги. Інформаційна безпека – це стан захищеності суспільства, держави, особистості, стан захищеності інформаційних ресурсів, які забезпечують прогресивний розвиток життєво важливих сфер суспільства.

Сфера інформаційних технологій потребує дуже багато спеціалістів. Інформаційні технології зараз застосовуються у всіх сферах життя. Через це виникає необхідність спеціалістів різних напрямків зі знанням специфічних технологій. Розробка системи підтримки прийняття рішень при аналізі інформаційного забезпечення безпеки даних в залежності від потреб ринку інформаційних технологій допоможе покращити якість аналізу можливих загроз втрати інформації.

Безпека та захист в інформаційних системах мають складатися з урахуванням комплексного підходу до побудови системи захисту, що означає об'єднання в єдиний комплекс необхідних заходів та засобів захисту даних на всіх рівнях системи інформаційного забезпечення. Потрібно щоб система інформаційної безпеки була спрямована на запобігання втраті інформації, її перекручення, незахищеного доступу та незаконного її застосування у проектуванні, впровадженні та експлуатації інформаційних підсистем [1]. Саме це визначає що створення та підтримка системи захисту та безпеки бази даних є одним із найважливіших аспектів при розробці та функціонуванні будь-якої інформаційної системи.

Що стосується даних в системах баз даних, то вони мають зберігатися з гарантуванням конфіденційності та безпеки. Інформаційні дані не можуть бути загублені або викрадені. Під безпекою даних у базі розуміють захист їх від випадкового або запланованого доступу до них осіб, які не мають таких прав, від несанкціонованого викриття, зміни або видалення.

Можливі негативні впливи різноманітних видів інформаційної безпеки, тобто захищеність даних та підтримуючої інфраструктури від випадкових чи запланованих природних або штучних впливів, які можуть приносити збитки їхнім власникам або користувачам. Інформаційна безпека також означає рівень захищеності інформаційного середовища суспільства, який забезпечує його формування, використання та розвиток в інтересах громадян, організацій, держави і нейтралізації негативних наслідків інформатизації суспільства.

Безпека в інформаційних даних в різних сферах суспільства має окрему специфіку. У політиці інформаційна безпека визначається інформаційно-аналітичною діяльностю дипломатичних представництв і зовнішньо-економічних відносин [2]. В економічних сферах інформаційна безпека стосується захисту інформаційних даних у системах банкінгу та мережових зв'язків, а особливо захисту конфіденційної економічної інформації та даних від незаконного доступу. Поняття саме інформаційної безпеки тісно пов'язане із інфомраційною загрозою.

До стратегії інформаційної безпеки повинні входити забезпечення захисту вже напрацьованих об'єктів інтелектуальної власності, що мають комерційну цінність та запобігання втрати таких об'єктів в майбутньому.

Забезпечення інформаційної безпеки має бути зорієнтоване, по-перше, на запобігання ризиків, та вже потім на знищення їх наслідків. Саме прийняття запобіжних заходів із забезпечення конфіденційності, цілісності, а також доступності даних і є найдієвішим підходом у здійсненні системи забезпечення інформаційної безпеки.

Будь-який витік інформації може призвести до серйозних проблем для підприємства, від значних фінансових збитків до повної ліквідації. Звичайно, проблема витоків секретних даних з'явилася не у наш час, промислове шпигунство і переманювання кваліфікованих фахівців існували ще й до епохи комп'ютеризації. Але саме з появою персональних комп'ютерів та інтернету виникли нові прийоми незаконного отримання інформації.

Найчастіше під атаки зловмисників потрапляють з компаній документи фінансового характеру, технологічні і конструкторські розробки, логіни і паролі для входу в мережу інших організацій. Але серйозної шкоди може завдати і витік персональних даних співробітників.

Хоча кількість загроз постійно зростає, з'являються все нові і нові віруси, збільшується інтенсивність і частота інформаційних атак, розробники засобів захисту інформації теж оновлюють та створюють системи забезпечення безпеки даних. На кожну загрозу розробляється нове захисне програмне забезпеченню або вдосконалюється вже наявне [3].

Захист інформації повинен здійснюватись комплексно, відразу по декількох напрямках. Чим більше методів буде задіяно, тим менше ймовірність виникнення загроз і витоку інформації, тим стійкіше положення усієї інформаційної системи та цілісності підприємства, у якому така система буде впроваджена.

Розроблений програмний продукт буде дуже корисним інструментом у області інформаційного забезпечення безпеки даних тому що система відстежує найбільш небезпечний період часу, під час якого здійснюється велика кількість атак та спроб несанкціонованого доступу, та видає спеціалісту із забезпечення безпеки рекомендації щодо уникнення таких атак. Програма може бути застосована у всіх сферах життєдіяльності людини, де може існувати небезпека для інформаційних даних, тому вона є універсальною для багатьох підприємств.

В даній роботі також було проаналізовано можливі загрози забезпеченню безпеки даних, спроби несанкціонованого доступу до системи та видача рекомендацій користувачу щодо зменшення кількості атак на систему.

1. ПОСТАНОВКА ЗАДАЧІ

Задача забезпечення інформаційної безпеки даних це забезпечення стану захищеності систем обробки і зберігання даних, при якому збережено конфіденційність, доступність і цілісність інформації, або комплекс заходів, спрямованих на забезпечення захищеності інформації від несанкціонованого доступу, використання, оприлюднення, руйнування, внесення змін, ознайомлення, перевірки запису чи знищення (у цьому значенні частіше використовують термін «захист інформації»).

Проблема інформаційної безпеки розглядається у трьох основних аспектах, таких як безпосередньо захист інформації, контроль за відповідним інформаційним простором та достатнє інформаційне забезпечення державних і недержавних органів, громадських, приватних організацій та підприємств [4].

Забезпечення безпеки інформації або інформаційних даних у системах визначається відсутністю неприпустимого ризику, який пов'язаний з витоком інформації через технічні канали, незаконними і ненавмисними діями на дані або на інші ресурси автоматизованої інформаційної системи, що використовуються в продукті.

Метою реалізації інформаційної безпеки будь-якого об'єкта є розробка продукту забезпечення інформаційної безпеки даного об'єкту. Для побудови та ефективної експлуатації такого програмного продукту забезпечення інформаційної безпеки необхідно, перш за все, щоб програмний продукт забезпечував авторизацію спеціалістів із забезпечення безпеки даних, можливість роботи з базами даних, можливість вибору методу збереження баз даних, аналізування загроз безпеки інформації, візуальне представлення

результату роботи системи.

Політика інформаційної безпеки повинна відображати такі етапи створення засобів захисту інформації, як визначення інформаційних і технічних ресурсів, що підлягають захисту, виявлення потенційно можливих загроз і каналів витоку інформації, надання оцінки ризикам втрати інформації за наявної загроз, визначення вимог до системи захисту, безпосередньо сам вибір засобу захисту даних та його характеристик, впровадження та подальша підтримка обраних заходів, способів та засобів захисту, здійснення контролювання цілісності і управління системою захисту [5].

Розроблена програма повинна проаналізувати різні підходи до проблеми забезпечення інформаційної безпеки залежно від кількості спроб несанкціонованого доступу та часу, коли такі спроби були зареєстровані.

Метою розробки також є створення програмного продукту, який забезпечує цілісність та захищеність інформації та даних при спробах несанкціонованого доступу до них шляхом надання спеціалісту із безпеки рекомендацій відповідно до аналізу кількості атак відносно часу.

Програмний продукт повинен забезпечувати:

— авторизацію користувача;

— можливість роботи з базою даних;

— можливість вибору методу збереження бази даних;

— можливість аналізу загроз безпеки даних;

— візуалізацію інструментів вибірки;

— візуалізацію результату аналізу загроз.

Система повинна відловлювати помилки та виводити повідомлення для користувача у разі виникнення несподіваних умов роботи.

2. АНАЛІЗ ПРОБЛЕМИ ІНФОРМАЦІЙНОГО ЗАБЕЗПЕЧЕННЯ БЕЗПЕКИ ДАНИХ. ПІДСИСТЕМА АНАЛІЗУ

В даній роботі захист інформації передбачає систему заходів, спрямованих на недопущення несанкціонованого доступу до інформації, несанкціонованої її модифікації, втрати, знищення, порушення цілісності тощо а також аналіз можливих атак на систему.

2.1 Загальні поняття

Інформаційна безпека — це стан захищеності систем обробки і зберігання даних, при якому забезпечено конфіденційність, доступність і цілісність інформації, використання й розвиток в інтересах громадян або комплекс заходів, спрямованих на забезпечення захищеності інформації особи, суспільства і держави від несанкціонованого доступу, використання, оприлюднення, руйнування, внесення змін, ознайомлення, перевірки запису чи знищення (у цьому значенні частіше використовують термін «захист інформації») [6].

RAID (Redundant Array of Independent Disks) — технологія віртуалізації даних, яка об'єднує декілька дисків в логічний елемент для надмірності і підвищення продуктивності.

Система безпеки — це сукупність засобів і методів підтримки безпечного стану об'єкта, попередження, виявлення та ліквідації загроз стану та середовища існування даних та інформації.

Хмарне сховище — це служба, що дозволяє передавати дані в зовнішні системи через мережу. Виділене місце на численних серверах постачальника послуг, система зберігання файлів децентралізована, тобто два або більше

файлів можуть бути на абсолютно різних серверах. Плюси і переваги хмарних сховищ: файли доступні скрізь, де є інтернет, ви можете отримати доступ з телефону або з ПК; економія місця або розширення пам'яті, фотографії та відео можна зберігати в хмарі для економії місця; велика швидкість передачі файлів, завдяки широкій географічній мережі серверів; надійність зберігання: навіть якщо один сервер вийде з ладу - на інших серверах є копії даних.

Система контролю і управління доступом (СКУД) — сукупність програмно-апаратних технічних засобів контролю і засобів управління, що мають на меті обмеження і реєстрацію входу-виходу об'єктів на заданій території через «точки проходу». Виконує функцію відкритої системи, що забезпечує технологію безпеки, яка дозволяє або забороняє доступ до певних типів даних, засновану на ідентифікації суб'єкта, якому потрібен доступ, і об'єкта даних, що є метою доступу.

Веб-сервер — сервер, який приймає запити від клієнтів, зазвичай веб-браузерів, і видає їм відповіді, як правило, разом з інтернет сторінкою, зображенням, файлом, медіа-потоком або іншими даними.

Інформаційна атака — це сукупність навмисних дій зловмисника, спрямованих на порушення одного з трьох властивостей інформації - доступності, цілісності або конфіденційності. Виділяють три етапи реалізації атак: етап підготовки і збору інформації про об'єкт атаки, етап реалізації атаки, етап усунення слідів і інформації про зловмисника.

Файрвол — програмний або програмно-апаратний елемент комп'ютерної мережі, що здійснює контроль і фільтрацію переданого через нього мережевого трафіку відповідно до заданих правил. Серед завдань, які вирішують міжмережеві екрани, основним є захист сегментів мережі або окремих хостів від несанкціонованого доступу з використанням вразливих місць в протоколах мережевий моделі або в програмному забезпеченні, встановленому на комп'ютерах мережі. Міжмережеві екрани пропускають або забороняють трафік, порівнюючи його характеристики з заданими шаблонами. Найбільш поширене місце для установки міжмережевих екранів - межа периметра

локальної мережі для захисту внутрішніх хостів від атак ззовні. Однак атаки можуть починатися і з внутрішніх вузлів - в цьому випадку, якщо атакується хост розташований в тій же мережі, трафік не перетне кордон мережевого периметра, і міжмережевий екран не буде задіяний. Тому в даний час міжмережеві екрани розміщують не тільки на кордоні, але і між різними сегментами мережі, що забезпечує додатковий рівень безпеки.

База даних — це організована структура, призначена для зберігання, зміни і обробки взаємозалежної інформації, переважно великих обсягів. Бази даних активно використовуються для динамічних сайтів зі значними обсягами даних - часто це інтернет-магазини, портали, корпоративні сайти.

Система управління базами даних — це комплекс програмних засобів, необхідних для створення структури нової бази, її наповнення, редагування вмісту і відображення інформації. У свою чергу, для зручності роботи з СУБД використовуються спеціальні веб-додатки, які дозволяють за допомогою графічного інтерфейсу виконувати адміністрування сервера баз даних, запускати спеціальні команди, а також працювати з контентом таблиць і баз даних - дії, які при відсутності веб-додатки підлягають виконанню засобами консолі.

Забезпечення безпеки — це комплекс дій, спрямованих на забезпечення нормального перебігу процесу, без виникнення будь-яких ризиків, надзвичайних ситуацій, а також відсутність шкідливого впливу на зовнішнє середовище.

2.2 Огляд існуючих систем інформаційного забезпечення безпеки даних

Наразі існує немало комплексів технічних і організаційних заходів, за допомогою яких забезпечується захист інформаційних ресурсів від несанкціонованого використання. Інформаційна безпека є важливий елементом

в IT-структурі будь-якої організації [7]. Належне її забезпечення відповідає за цілісність інформаційної структури підприємства, збереження всіх її даних та їх конфіденційність.

Серед існуючих програмних засобів для підтримки та забезпечення безпеки даних можна виділити декілька продуктів на ринку.

Міжмережеві екрани вже являють собою не прості пакетні фільтри, а складні багатофункціональні комплекси по забезпеченню безпеки - NGFW (Next Generation Firewall), здійснюючи на найвищому рівні захист від атак, вірусних епідемій, бот-мереж і невідомих видів загроз (загроз «нульового дня). У всіх пристроях або програмних забезпеченнях застосовується різноманітний набір потужних механізмів знаходження сучасних видів загроз, це використання світових репутаційних баз файлів та посилань, і останніх розробок в сфері машинного навчання, і запуск файлів з невідомим або підозрілим вмістом у віртуальних середовищах, і використання поведінкового аналізу, заснованого на штучному інтелекті та багато іншого.

Антивірусний захист — це вирішення по захисту робочих станцій та серверного обладнання від атак і шкідливого програмного забезпечення. Новітні рішення застосовують водночас декілька шляхів виявлень: сигнатурний аналіз (неактуальна на сьогодення технологія), репутаційний аналіз, поведінковий аналіз і використання «пісочниць» (аналіз виконання підозрілого вмісту в віртуальному середовищі). Для досягнення максимальної ефективності, дані рішення не тільки повинні мати централізовану систему управління та збору даних про події, але також інтегруватися в загальну систему, що дозволяло б їм сповіщати інші модулі компанії (міжмережевий екран, рішення щодо захисту пошти і web-доступу користувачів) про знайдені шкідливі елементи, а також централізовано отримувати такі дані від них.

Ще однією критично важливою функцією, є можливість надавати дані для розслідування інцидентів (повного комплексного ретроспективного аналізу), які б могли допомогти у виявленні, як джерел зараження (якщо таке мало місце), так і слабких місць в інфраструктурі інформаційної безпеки.

SIEM (Security information and event management) — рішення по об'єднанню подій безпеки з різних пристроїв і пошуку інцидентів безпеки на основі аналізу (кореляції) цих даних. Такі рішення є вендоронезалежними, і як наслідок, дають можливість максимально інтегруватися в абсолютно будь-яку інформаційну інфраструктуру. Крім метоів аналізу і кореляції, SIEM системи також надають потужні інструменти звітності та сповіщення, дозволяючи робити видимими навіть складні приховані атаки [8].

Системи запобігання витоків інформації (DLP) — система запобігання витоків інформації яка є багаторівневим комплексом із захисту секретних даних, від випадкового або навмисного незаконного поширення, або їх крадіжки.

Внаслідок успішного проходження попереднього процесу машинного навчання, дана система може виявити і запобігти навіть добре сховані і розтягнуті за часом спроби несанкціонованого доступу.

Системи автоматизації правил безпеки — проводження глобальних оптимізацій баз правил, перегляд роботи сервісів і бізнес-процесів. Основною метою такої системи є допомога в наведенні порядку в політиках міжмережевих екранів. Система дає рекомендації із видалення невикористовуваних правил, або щодо обмеження правил із занадто широкими доступами; показує правила, які порушують корпоративні політики; вказує оптимальне місце при створенні нових політик та багато іншого. У додаток система проводить постійний аналіз всіх внесених змін в політики, може порівнювати різні версії політик, а також чи відповідають вони світовим стандартам і кращим світовим практикам.

Системи управління обліковими даними (Identity Management) — це надання централізованого управління обліковими даними, такими як запити доступу користувачем на порталі самообслуговування, закінчуючи автоматичною видачею або вилученням необхідних прав (що базуються на рольовій основі) в регламентованих інформаційних системах. Можливість представлення звітів про збережені права зацікавленим користувачем, або їх групою, а також атаки на системи за необхідний період часу. Система замінює

ручну видачу потрібних прав користувачам чітким, заздалегідь прописаним автоматизованим процесом, що унеможливлює помилки, викликані «людським фактором», а також умисне порушення встановлених на підприємстві процедур отримання / вилучення прав доступу.

Системи аутентифікації користувачів — це впровадження систем строгої двухфакторної аутентифікації, яка полягає як у застосуванні спеціалізованих пристроїв (токенів) для зручності та надійності аутентифікації користувачів, так і системи централізованого управління життєвим циклом токенів.

Enterprise Resource Planning — дозволяє управляти бізнес процесами, отримувати достовірні дані про роботу компанії в режимі реального часу і відстежувати ключові показники діяльності для оперативного прийняття управлінських рішень із забезпечення безпеки, забезпечує конкурентоспроможність системи і дозволяє підвищувати її ефективність.

Забезпечення інформаційної безпеки від Microsoft — сервіси інформаційної безпеки Microsoft, вбудовані інструменти та елементи управління яких відповідають корпоративним вимогам, забезпечують надійний контроль даних, можливість класифікувати і маркувати файли, відстежувати їх використання і при необхідності змінювати рівні доступу, а також запобігати втраті або витоку конфіденційної інформації. Сьогодні забезпечення інформаційної безпеки для бізнесу має формувати багатогранний підхід, який забезпечить постійну і надійну IT безпеку, а також допоможе виявити ранні сигнали порушень і зреагувати на них до того, як загроза завдасть шкоди.

Захист від втрати даних — служба дозволяє визначати, відстежувати і автоматично захищати конфіденційну інформацію від розголошення. Сервіс запобігає витоку конфіденційних відомостей за допомогою глибокого аналізу вмісту, а не простого сканування тексту. Для виявлення контенту, що порушує політики захисту від втрати даних, при поглибленому аналізі застосовуються ключові слова, словникові збіги, оцінка регулярних виразів, структури і контрольних сум фрагментів документів і інші методи.

Переваги:

— постійне відстеження порушень правил безпеки;

— налаштування індивідуальних параметрів конфіденційності;

— наявність шаблонів політик захисту даних;

— можливість перевірки роботи служби в тестовому режимі.

Центр безпеки та відповідності вимогам — єдиний портал захисту даних в Microsoft Office, який забезпечує повний контроль над особистими даними і дозволяє управляти правами доступу до них.

Центр безпеки надає можливість управляти архівними поштовими скриньками, налаштовувати політику виявлення електронних даних, політику зберігання і видалення документів в Exchange Online і SharePoint Online, а також переглядати деталізовані звіти про активність.

Переваги:

— запобігання випадковому розголошенню конфіденційних даних;

— відстеження джерел витоку інформації;

— очищення даних з мобільних пристроїв у випадках втрати або крадіжки;

— архівація повідомлень поштової скриньки;

— автоматизоване видалення документів.

Управління загрозами — набір інструментів і панелей моніторингу даних, які використовуються для пошуку і запобігання загрозам. Служба відстежує сигнали з різних джерел, включаючи дії користувачів, перевірку справжності, електронну пошту, скомпрометовані комп'ютери і інциденти безпеки. Дані аналізуються і відображаються, щоб адміністратори безпеки змогли вчасно зреагувати на загрозу, спрямовану проти користувачів та інтелектуальної власності.

Переваги:

— своєчасне реагування на фішинг і шкідливі програми;

— виявлення цільових атак;

— пошук загроз за індикаторами;

— докладні відомості про шкідливі програми;

— статистика загроз по галузям в режимі реального часу.

Azure Information Protection — хмарне рішення, яке дозволяє організаціям класифікувати, маркувати і захищати документи та електронні листи. Azure Information Protection забезпечує постійний і надійний захист в будь-який час незалежно від того, де дані зберігаються або для кого публікуються. Це дозволяє швидко ідентифікувати необхідну інформацію [9].

Переваги:

— повний контроль і управління доступом до даних;

— захист інформації на серверах хмарного сервісу;

— захист інформації без прив'язки до місця зберігання;

— захищені спільні дані;

— інтуїтивно зрозумілий інтерфейс.

Azure Active Directory — це хмарна служба управління призначеними для користувача каталогами, посвідченнями та авторизованим доступом до даних підприємства, яка забезпечує уніфікований контроль і управління за службами ідентифікації, об'єднує в собі сервіси авторизації, розгортання і моніторингу політик безпеки.

Advanced Threat Analytics — є автоматизованою службою виявлення кіберзагроз та проведення поведінкового аналізу підозрілих дій в IT-середовищі компанії. За допомогою Advanced Threat Analytics є можливість проводити моніторинг підозрілих дій, і в разі загрози, визначити точку несанкціонованого доступу до ресурсів підприємства; налаштування автоматичних поштових повідомлень про погрози специфічним групам користувачів; інтегрування ATA в SIEM для поліпшення корпоративної безпеки.

Переваги:

— комплексне виявлення атак;

— поведінкова аналітика;

— поштові повідомлення про погрози;

— мобільність;

— інтеграція з SIEM.

2.3 Характеристики загроз інформаційної безпеки

Вибір інструментів забезпечення безпеки даних для аналізу є одним з найважливіших етапів створення програмного продукту для підтримки прийняття рішень при аналізі інформаційного забезпечення безпеки даних енергетичних процесів та систем.

Розвиток комп’ютерних технологій і їх використання в багатьох сферах життя є на сьогодні одним з головних факторів її ефективності. Проте прогрес в інформаційно-технічній сфері створив і потенційні загрози у вигляді розроблення нових та удосконалення вже відомих методів наукового шпигунства, котрі дозволяють швидко знаходити в комп’ютері необхідні відомості. Застосування в управлінській технології електронного документообігу в цілому створює можливість несанкціонованого доступу до комунікацій. Відтік інформації з обмеженим доступом, що має реальну цінність, напряму пов’язаний з очікуваними результатами. І саме це може завдати значної шкоди інтересам власника інформації.

Загрози інформаційної (комп'ютерної) безпеки це різні дії, які можуть привести до порушень інформаційної безпеки. Іншими словами, це потенційно можливі події / процеси або дії, які можуть завдати шкоди інформаційних та комп'ютерних систем.

Загрози ІБ можна розділити на два типи: природні і штучні. До природних відносяться природні явища, що не залежать від людини, наприклад, урагани,

повені, пожежі і т.д. Штучні загрози залежать безпосередньо від людини і можуть бути навмисні і ненавмисні [10]. Ненавмисні загрози виникають через необережність, неуважність і незнання. Прикладом таких загроз може бути установка програм, які не входять до числа необхідних для роботи, в подальшому порушують роботу системи, що і призводить до втрати інформації. Навмисні загрози, на відміну від попередніх, створюються спеціально. До них можна віднести атаки зловмисників як ззовні, так і зсередини компанії. Результат цього виду загроз - величезні втрати компанією грошових коштів та інтелектуальної власності.

Залежно від різних способів класифікації всі можливі загрози інформаційної безпеки можна розділити на наступні основні підгрупи:

— небажаний контент;

— несанкціонований доступ;

— витік інформації;

— втрата даних;

— шахрайство;

— кібервійни;

— кібертероризм.

Небажаний контент включає в себе не тільки шкідливі програми, потенційно небезпечні програми і спам, які безпосередньо створені для того, щоб знищити або вкрасти інформацію, але і сайти, які заборонені законодавством, або небажані сайти, що містять інформацію, яка не відповідає сподіванням споживача.

Несанкціонований доступ це перегляд інформації співробітником, який не має дозволу користуватися даною інформацією, шляхом порушення посадових повноважень. Несанкціонований доступ призводить до витоку інформації. Залежно від того, яка інформація і де вона зберігається, витік даних може організовуватися різними способами, а саме через атаки на сайти, злом програм, перехоплення даних по мережі, використання несанкціонованих

програм.

Витік інформації в залежності від того, чим вона була викликана, може поділятися на навмисну і випадкову. Випадкові витоки відбуваються через помилки обладнання, програмного забезпечення і людини. А умисні, на відміну від випадкових, організовуються навмисно, з метою отримати доступ до даних, завдати шкоди.

Втрату даних можна вважати однією з основних загроз інформаційній безпеці. Порушення цілісності інформації може бути викликано несправністю обладнання або навмисними дії користувачів, будь то вони співробітниками або зловмисниками.

Не менш небезпечною загрозою є фрод (шахрайство з використанням інформаційних технологій). До шахрайства можна віднести не тільки маніпуляції з кредитними картами (кардинг) і злом онлайн-банку, але і внутрішній фрод. Метою цих економічних злочинів є обхід законодавства, політики, нормативних актів компанії, привласнення майна.

Щорічно по всьому світу зростає терористична загроза поступово переходячи у віртуальний простір. На сьогоднішній день нікого не дивує можливість атак на АСУ різних підприємств. Але подібні атаки не проводяться без попередньої розвідки, для чого і потрібен кібершпіонаж, який допоможе зібрати необхідні дані. Існує також таке поняття, як інформаційна війна, яка відрізняється від звичайної війни тільки тим, що в якості зброї виступає ретельно підготовлена інформація.

Порушення інформаційної безпеки може бути викликано як спланованими діями зловмисника, так і недосвідченістю співробітника. Користувач повинен мати хоч якесь поняття про ІБ, шкідливий програмному забезпеченні, щоб своїми діями не завдати шкоди компанії і самому собі. Щоб пробитися через захист і отримати доступ до потрібної інформації зловмисники використовують слабкі місця і помилки в роботі програмного забезпечення, веб-додатків, помилки в конфігураціях файрволов, прав доступу, вдаються до

прослуховування каналів зв'язку і використання клавіатурних шпигунів.

Втрата інформації може бути обумовлена не тільки зовнішніми атаками зловмисників і недбалістю співробітників, але і працівниками компанії, які зацікавлені в отриманні прибутку в обмін на цінні дані організації, в якій працюють або працювали.

Те, як буде проводитися атака, залежить від типу інформації, її розташування, способів доступу до неї і рівня захисту. Якщо атака буде розрахована на недосвідченість жертви, то можливе використання спам розсилок.

Оцінювати загрози інформаційної безпеки необхідно комплексно, при цьому методи оцінки будуть відрізнятися в кожному конкретному випадку. Наприклад, щоб виключити втрату даних через несправність обладнання, потрібно використовувати якісні комплектуючі, проводити регулярне технічне обслуговування, встановлювати стабілізатори напруги. Далі слід встановлювати і регулярно оновлювати програмне забезпечення. Окрему увагу потрібно приділити захисному ПО, бази якого повинні оновлюватися щодня:

— захист від небажаного контенту (антивірус, антиспам, веб-фільтри, анти-шпигуни);

— фаєрволи і системи виявлення вторгнень IPS;

— IDM;

— PUM;

— захист веб-додатків;

— анти-ддос;

— WAF;

— аналіз вихідного коду;

— антіфрод;

— захист від націлених атак;

— SIEM;

— системи виявлення аномального поведінки користувачів;

— захист АСУ;

— захист від витоків даних;

— DLP;

— шифрування та захист мобільних пристроїв;

— резервне копіювання та системи відмовостійкості.

Навчання співробітників компанії основним поняттям інформаційної безпеки і принципам роботи різних шкідливих програм допоможе уникнути випадкових витоків даних, виключити випадкову установку потенційно небезпечних програм на комп'ютер. Також в якості запобіжного заходу від втрати інформації слід робити резервні копії. Для того щоб стежити за діяльністю співробітників на робочих місцях і мати можливість виявити зловмисника, слід використовувати DLP-системи.

Загрози інформаційної безпеки можуть бути класифіковані за різними ознаками. За аспектом інформаційної безпеки, на який спрямовані загрози:

— загрози конфіденційності (неправомірний доступ до інформації). Загроза порушення конфіденційності полягає в тому, що інформація стає відомою тому, хто не має в повноважень доступу до неї. Вона має місце, коли отримано доступ до деякої інформації обмеженого доступу, що зберігається в обчислювальній системі чи переданої від однієї системи до іншої. У зв'язку з загрозою порушення конфіденційності, використовується термін «витік». Подібні загрози можуть виникати внаслідок «людського фактора» (наприклад, випадкове делегування того чи іншого користувачеві привілеїв іншого користувача), збоїв роботі програмних і апаратних засобів. До інформації обмеженого доступу належить державна таємниця і конфіденційна інформація (комерційна таємниця, персональні дані, професійні види таємниця: лікарська, адвокатська, банківська, службова, нотаріальна, таємниця страхування, слідства і судочинства, листування, телефонних переговорів, поштових відправлень, телеграфних або інших повідомлень (таємниця зв'язку), відомості про сутність

винаходу, корисної моделі чи промислового зразка до офіційної публікації (ноу-хау) і ін.);

— загрози цілісності (неправомірна зміна даних). Загрози порушення цілісності це загрози, пов'язані з імовірністю модифікації тієї чи іншої інформації, що зберігається в інформаційній системі. Порушення цілісності може бути викликано різними факторами - від навмисних дій персоналу до виходу з ладу обладнання;

— загрози доступності (здійснення дій, що унеможливлюють чи утруднюють доступ до ресурсів інформаційної системи). Порушення доступності є створення таких умов, при яких доступ до послуги або інформації буде або заблокований, або можливий за час, який не забезпечить виконання тих чи інших бізнес-цілей.

По розташуванню джерел загроз:

— внутрішні (джерела загроз розташовуються усередині системи);

— зовнішні (джерела загроз знаходяться поза системою).

За розмірами завданої шкоди:

— загальні (нанесення збитку об'єкту безпеки в цілому, заподіяння значної шкоди);

— локальні (заподіяння шкоди окремим частинам об'єкта безпеки);

—приватні (заподіяння шкоди окремим властивостям елементів об'єкта безпеки).

За ступенем впливу на інформаційну систему:

— пасивні (структура і зміст системи не змінюються);

— активні (структура і зміст системи піддається змінам).

За природою походження:

— природні (об'єктивні), викликані впливом на інформаційне середовище об'єктивних фізичних процесів або стихійних природних явищ, що не залежать від волі людини;

— штучні (суб'єктивні), викликані впливом на інформаційну сферу людини. Серед штучних загроз в свою чергу виділяють: ненавмисні (випадкові) загрози , тобто помилки програмного забезпечення, персоналу, збої в роботі систем, відмови обчислювальної і комунікаційної техніки і навмисні (умисні) загрози, такі як неправомірний доступ до інформації, розробка спеціального програмного забезпечення, використовуваного для здійснення незаконного втручання, розробка та поширення вірусних програм і т.д. Навмисні загрози обумовлені діями людей. Основні проблеми інформаційної безпеки пов'язані перш за все з навмисними погрозами, так як вони є головною причиною злочинів і правопорушень.

Носіями загроз безпеці інформації є джерела загроз. Як джерела загроз можуть виступати як суб'єкти (особистість), так і об'єктивні прояви, наприклад, конкуренти, злочинці, корупціонери, адміністративно-управлінські органи. Джерела загроз переслідують при цьому наступні цілі: ознайомлення з охоронюваними відомостями, їх модифікація в корисливих цілях і знищення для нанесення прямих матеріальних збитків.

Всі джерела загроз інформаційній безпеці можна розділити на три основні групи:

— обумовлені діями суб'єкта (антропогенні джерела), суб'єкти, дії яких можуть призвести до порушення безпеки інформації, дані дії можуть бути кваліфіковані як умисні або випадкові злочину. Джерела, дії яких можуть призвести до порушення безпеки інформації можуть бути як зовнішніми так і внутрішніми. Дані джерела можна спрогнозувати, і вжити адекватних заходів;

— обумовлені технічними засобами (техногенні джерела), ці джерела загроз менш прогнозовані, безпосередньо залежать від властивостей техніки і тому вимагають особливої уваги. Дані джерела загроз інформаційній безпеці, також можуть бути як внутрішніми, так і зовнішніми;

— стихійні джерела, дана група об'єднує обставини, що становлять непереборну силу (стихійні лиха або інші обставини, які неможливо

передбачити або запобігти або можливо передбачити, але неможливо запобігти), такі обставини, які носять об'єктивний і абсолютний характер, поширюється на всіх. Такі джерела загроз абсолютно не піддаються прогнозуванню і, тому заходи проти них повинні застосовуватися завжди. Стихійні джерела, як правило, є зовнішніми по відношенню до захищається і під ними, як правило, розуміються природні катаклізми.

Несанкціонований доступ — доступ до інформації в порушення посадових повноважень співробітника, доступ до закритої для публічного доступу інформації з боку осіб, які не мають дозволу на доступ до цієї інформації. Також несанкціонованим доступом в окремих випадках називають отримання доступу до інформації особою, яка має право на доступ до цієї інформації в обсязі, що перевищує необхідний для виконання службових обов'язків. Несанкціонований доступ може привести до витоку інформації. Несанкціонований доступ до інформації (НСД) — доступ до інформації, що порушує правила розмежування доступу з використанням штатних засобів, що надаються засобами обчислювальної техніки або автоматизованими системами.

2.4 Інструмент для забезпечення безпеки даних Firewall

У світі стрімко зростаючого росту використання технологій, обійтися без захисту від шкідливих програм неможливо. За статистикою комп'ютер з виходом в інтернет, що не має захисту, залишається зараженим протягом двох хвилин. Саме з цієї причини і була розроблена така корисна програма, як фаєрвол, яка захищає комп'ютер від вірусів. Фаєрвол — це програма, назва якої з англійської перекладається, як «палаюча стіна», вона встановлює перепону між комп'ютером і надходженою в нього інформацією. Існує еквівалент цієї програми — брандмауер. Вона відображає суть і призначення цього механізму, тому як завдяки функціональним можливостям ця програма підвищує ступінь захисту комп'ютера. Це своєрідна стіна з вогню, яка пропускає через себе потік інформації з інтернету, очищаючи його від непотрібного і шкідливого сміття.

Отже, комп'ютер, на якому працює фаєрвол, завжди знаходиться під захистом [11].

Поряд із захистом від шкідливих файлів брандмауер також запобігає передачі шкідливих програм на інші комп'ютери або в інтернет. Це вбудована в операційну систему Windows програма, мета якої перешкоджати проникненню шкідливих файлів, вірусів, троянів, що надходять в неї через інтернет. Фаєрвол був розроблений і адаптований і для інших операційних систем, наприклад, для ОС Linux.

При установці системи Windows фаєрвол буде за замовчуванням включений. Однак його також можна відключити, якщо він перешкоджає коректній роботі програми або завантаження файлів з інтернету. Firewall здатний блокувати підключення користувача до програм, яких немає в списку дозволених. Таким чином, кожна Невизначена програма буде заблокована фаєрволом автоматично. Є можливість налаштувати роботу брандмауера відповідно до особистих переваг користувача, наприклад, так щоб при блокуванні фаєрволом програми спливало відповідне повідомлення. Безумовно, фаєрвол — це необхідна і корисна програма для будь-якого комп'ютера. Вона допомагає запобігти незаконному вторгненню в систему і тримати її в безпеці, запобігаючи відправку шкідливих файлів на інші пристрої. Також, крім вбудованого в систему брандмауера фахівці рекомендують встановлювати інші антивірусні програми, оскільки фаєрвол не завжди справляється з напором вірусів. Принципом роботи цієї системи є інструмент, який намагається визначити, які протоколи рівня 4 передаються даними передавальним пристроєм по IP. Firewall намагається визначити, які протоколи будуть блокувати маршрутизатор або брандмауер, і які вони передадуть на хости. Він працює за методом стрибків IP-адреси, як і зазвичай використовується програма Traceroute. Метод IP-закінчення включає в себе маніпулювання часом життя поля IP-заголовка для відображення всіх проміжних маршрутизаторів або переходів між хостом сканування і цільовим хостом. У Firewall сканування відправляється зі значенням TTL на один стрибок вище, ніж у цільового хоста.

Грунтуючись на результатах сканування, Firewall може визначити, які порти відкриті. Схематичне зображення роботи Firewall (рисунок 2.1).

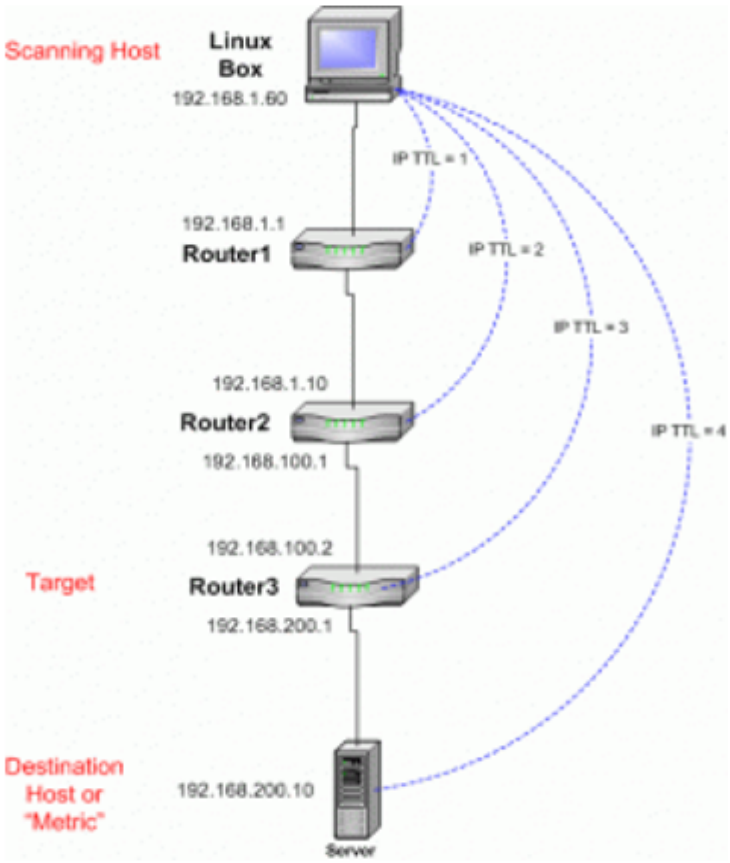

Рисунок 2.1 — Схема роботи інструменту Firewall

2.4.1 Апаратний фаєрвол

Встановлюється як окремий пристрій і налаштовується так, що він працює в якості “двері” в вашу мережу. Для роботи такого фаервола, його слід встановлювати між локальною мережею і інтернетом. Перевага цього методу полягає необхідності для зловмисника спочатку «зламати» фаєрвол, перш ніж отримати прямий доступ до будь-яких мережевих ресурсів. Недолік цього методу полягає в необхідності придбання додаткових апаратних засобів.

Іноді апаратним фаєрволом називають окремий комп'ютер, виділений спеціально для цих цілей. На нього встановлюється необхідне програмне

забезпечення та дві мережеві карти (одна «дивиться» в вашу мережу, інша - в мережу інтернет). В даному випадку, вам досить буде низько продуктивного комп'ютера, так як зазвичай програмне забезпечення не вимогливо до ресурсів.

Переваги апаратних брандмауерів:

— Відносна простота розгортання і використання. Підключити, включити, поставити параметри через веб-інтерфейс. Втім, сучасні програмні міжмережеві екрани підтримують розгортання через ActiveDirectory, на яке теж не піде багато часу. Але, по-перше, не всі брандмауери підтримують ActiveDirectory, і, по-друге, не завжди на підприємстві використовується Windows.

— Розміри і енергоспоживання. Як правило, апаратні брандмауери мають більш скромні розміри і менше енергоспоживання. Не завжди енергоспоживання грає роль, а ось розміри системи іноді дуже важливі.

— Продуктивність. Зазвичай продуктивність у апаратного рішення вище. Хоча б тому, що апаратний міжмережевий екран займається тільки своєю безпосередньою функцією — фільтрацією пакетів. На ньому не запущені будь-які сторонні процеси і служби, як це часто буває у випадку з програмними брандмауерами. Ось уявіть, що ви організували програмний шлюз (з функціями брандмауера і NAT) на базі сервера з Windows Server. Навряд чи ви будете виділяти цілий сервер тільки під брандмауер і NAT. Це нераціонально. Швидше за все, на ньому будуть запущені і інші служби - той же AD, DNS, СКБД і поштові служби.

— Надійність. Вважається, що апаратні рішення більш надійні (саме через те, що на них рідко коли виконуються сторонні служби). Але можливо виділити окремий системник, встановити на нього FreeBSD (одна з найнадійніших в світі операційних систем) і налаштувати правила брандмауера. Надійність такого рішення буде не нижче, ніж у випадку з апаратним файрволом. Але таке завдання вимагає підвищеної кваліфікації адміністратора, саме тому раніше було відзначено, що апаратні рішення більш прості у

використанні.

2.4.2 Програмний фаєрвол

Програмним фаєрволом називається програмне забезпечення, яке встановлюється на комп'ютер, який необхідно захистити від мережевих погроз. Переваги цього типу полягають в більш простий налаштування і у відсутності додаткового обладнання. Недоліки програмних фаєрволів укладені в тому факті, що вони займають системні ресурси, і їх необхідно встановлювати на всіх робочих станціях і серверах мережі.

Переваги програмних брандмауерів:

— Вартість. Ціна програмного брандмауера зазвичай нижче заліза. За ціну середнього апаратного рішення можна захистити всю мережу програмним брандмауером.

— Можливість захисту мережі зсередини. Не завжди загрози походять тільки ззовні. Усередині локальної мережі є безліч загроз. Атаки можуть виходити з внутрішніх комп'ютерів. Ініціювати атаку може будь-який користувач LAN, наприклад, незадоволений підприємством. Як вже зазначалося, можна, звичайно, використовувати окремий апаратний маршрутизатор для захисту кожного окремого вузла, але на практиці таких рішень не зустрічаєтья. Це є дуже нераціонально.

— Можливість розмежування сегментів локальної мережі без виділення підмереж. У більшості випадків до локальної мережі підключаються комп'ютери різних відділів, наприклад, бухгалтерії, фінансового відділу, IT-відділу і т.д. Не завжди ці комп'ютери повинні взаємодіяти між собою. Як виконати розмежування. Перше рішення полягає в створенні декількох підмереж (наприклад, 192.168.1.0, 192.168.2.0 і т.д.) і налаштування належним чином маршрутизації між цими мережами. Не можна сказати, що рішення дуже складне, але все ж складніше, ніж використання програмного файрвола. Не

завжди можна виділити підмережі з тих чи інших причин. Друге рішення полягає у використанні брандмауера, призначеного саме для захисту мережі (не у всіх програмних міжмережевих екранах легко розмежувати). У цьому випадку навіть в найбільшій мережі виконується розмежування за лічені хвилини, не потрібно налаштувати маршрутизації.

— Можливість розгортання на існуючих серверах. Немає сенсу купувати ще одну «залізо», якщо є достатній комп'ютерний парк. Досить на одному з серверів розгорнути міжмережевий екран і налаштувати машрутізацію. Зазвичай обидві ці операції виконуються за допомогою графічного інтерфейсу брандмауера і реалізуються за допомогою декількох клацань мишею в потрібних місцях.

— Розширений функціонал. Як правило, функціонал програмних міжмережевих екранів ширше, ніж у їх апаратних аналогів. Так, деякі міжмережеві екрани надають функції балансування навантаження і тому подібні корисні речі, що дозволяють підвищити загальну безпеку системи обробки даних. Так, такі функції є не у всіх програмних брандмауерів, але ніщо і ніхто не заважає вибрати міжмережевий екран, відповідний потребам. Звичайно, такі функції є і у деяких апаратних комплексів. Наприклад, StoneGate — надає функціонал системи запобігання вторгнень, але вартість таких рішень не завжди підійде.

2.5 Обробка спроб несанкціонованого доступу

Захист інформації від несанкціонованого доступу є складовою частиною загальної проблеми забезпечення безпеки інформації. Заходи щодо захисту інформації від несанкціонованого доступу повинні здійснюватися взаємопов'язано із

заходами щодо спеціального захисту основних і допоміжних засобів обчислювальної техніки, засобів і систем зв'язку від технічних засобів розвідки

і промислового шпигунства [12].

Обробка спроб несанкціонованого доступу може бути забезпечена такими підсистемами:

— підсистема управління доступом;

— підсистема реєстрації та обліку;

— криптографічна підсистема;

— підсистема забезпечення цілісності.

У підсистемі управління доступом повинні здійснюватися ідентифікація і перевірка справжності суб'єктів доступу при вході в систему по паролю умовно-постійної дії, довжиною не менше шести букв і цифр. У підсистемі реєстрації та обліку повинна здійснюватися реєстрація входу (виходу) суб'єктів доступу в систему (з системи), або реєстрація завантаження і ініціалізації операційної системи і її програмної зупинки. Реєстрація виходу з системи або зупинки не проводиться в моменти апаратурного відключення АС. У параметрах реєстрації зазначаються: дата і час входу (виходу) суб'єкта доступу в систему (з системи) або завантаження (зупинки) системи; повинен проводитися облік всіх захищених носіїв інформації з допомогою їх маркування і з занесенням облікових даних в журнал (облікову картку). У підсистемі забезпечення цілісності повинна бути забезпечена цілісність програмних засобів систем захисту інформації, оброблюваної інформації, а також незмінність програмного середовища. При цьому:

— цілісність СЗІ перевіряється при завантаженні системи за наявністю імен (ідентифікаторів) компонент СЗІ;

— цілісність програмного середовища забезпечується відсутністю в АС коштів

розробки та налагодження програм;

— повинна здійснюватися фізична охорона пристроїв і носіїв інформації, що передбачає контроль доступу в приміщення АС сторонніх осіб, наявність надійних перешкод для несанкціонованого проникнення в приміщення АС і

сховище носіїв інформації;

— має проводитися періодичне тестування функцій СЗІ при зміні програмного середовища і персоналу АС за допомогою тест-програм, імітування спроб несанкціонованого доступу;

— повинні бути в наявності засоби відновлення СЗІ, що передбачають ведення двох копій програмних засобів СЗІ і їх періодичне оновлення і контроль працездатності.

Технологія оброблення інформації повинна бути здатною реалізовувати можливість виявлення спроб несанкціонованого доступу до інформації та процесів, які з цією інформацією пов’язані, а також забезпечити реєстрацію в системному журналі визначених політикою відповідної послуги безпеки подій. Для здійснення несанкціонованого доступу до інформації зловмисник не застосовує жодних апаратних або програмних засобів, що не входять до складу системи. Для захисту інформації від несанкціонованого доступу створюється система розмежування доступу до інформації. Завданням системи розмежування доступу є управління доступом користувачів до внутрішніх інформаційних ресурсів. Система має у своєму складі блоки ідентифікації і аутентифікації процесів, активованих певними користувачами; базу даних повноважень користувачів і керуючий блок. При необхідності виконання певних дій в користувач активує програму. Операційна система запускає процеси виконання програми, яким присвоюється ознака користувача. При зверненні процесів до ресурсів інформаційної системи (програм, файлів, пристроїв) система розмежування доступу визначає, в інтересах якого користувача ініційований процес, що вимагає звернення до ресурсів. Після аутентифікації процесу система розмежування доступу вибирає з бази даних відомості про повноваження користувача і порівнює їх з вимогами до повноважень, які визначені для ресурсу. Якщо повноваження користувача не менше потрібних, то відповідному процесу дозволяється виконати запитані дії по відношенню до ресурсу. Інакше йде відмова у виконанні операції і факт спроби порушення встановлених правил запам'ятовується в спеціальному журналі [13].

Для посилення стійкості систем до спроб несанкціонованого доступу до інформації використовується шифрування інформації на зовнішніх носіях, а також видаляються тимчасові файли після завершення інформаційних процесів.

На етапі експлуатації системи зловмисники можуть призвести наступні несанкціоновані дії по зміні технічної (апаратної) структури: підключення нештатних блоків, пристроїв або комп'ютерів; зміна зв'язків; заміна штатних блоків, пристроїв або комп'ютерів на відповідні структурні компоненти зі зміненими характеристиками; зміна режимів роботи пристроїв.

Несанкціонований доступ до апаратних і програмних засобів може бути виключений або суттєво утруднений при виконанні наступного комплексу заходів:

— охорона приміщень, в яких знаходяться апаратні засоби;

— протидія несанкціонованому входу в систему;

— протидія несанкціонованому підключенню обладнання;

—захист внутрішнього монтажу, засобів управління і комутації від несанкціонованого втручання.

У загальному випадку контрольований вхід в систему передбачає виконання таких дій:

— включення апаратних засобів;

— завантаження програмних засобів системи захисту інформації;

— ідентифікацію та аутентифікацію суб'єкта доступу;

— завантаження операційної системи;

— реєстрацію суб'єкта доступу;

— розблокування.

Для протидії несанкціонованому включенню апаратних засобів системи можуть застосовуватися такі методи:

— використання замків в блоках;

— дистанційне керування подачею живлення;

— блокування органів управління подачею живлення.

У захищених системах до завантаження штатної ОС може здійснюватися завантаження програмного блоку, контролюючого процес ідентифікації і аутентифікації, а також процес завантаження ОС. За рахунок відносної простоти і захищеності від несанкціонованих змін блок дозволяє здійснити ідентифікацію, аутентифікацію і довірене завантаження ОС. Ідентифікація та аутентифікація суб'єктів доступу можуть проводитися і засобами ОС після завершення її завантаження [14].

Комплекс заходів та засобів управління доступом до пристроїв повинен виконувати і функцію автоматичної реєстрації дій суб'єкта доступу. Журнал реєстрації подій може вестися як на автономній ЕОМ, так і в мережі. Періодично або при фіксації порушень протоколів доступу адміністратор переглядає журнал реєстрації з метою контролю дій суб'єктів доступу.

Завершується контрольований вхід в систему розблокуванням доступу і активуванням системи розмежування доступу для забезпечення авторизованого доступу до ресурсів системи.

Для відстеження загрози неконтрольованого підключення пристроїв використовуються наступні методи:

— перевірка особливостей пристроїв;

— використання ідентифікаторів пристроїв.

Контроль підключення пристроїв може здійснюватися за рахунок порівняння інформації при підключеному влаштуванні та відомостей про пристрої. До такої інформації відносяться типи пристроїв (блоків) та їх характеристики, кількість і особливості підключення зовнішніх пристроїв, режими роботи і інша інформація.

Ще більш надійним і оперативним методом контролю є використання спеціального коду-ідентифікатора пристрою. Цей код може генеруватися апаратними засобами, а може зберігатися в спеціальному запам'ятовуючому пристрої. Генератор може ініціювати видачу в контролюючий пристрій (в

обчислювальній мережі це може бути робоче місце адміністратора) унікального номера пристрою.

Для захисту від несанкціонованих дій щодо зміни монтажу, заміні елементів, переключенню комутуючих пристроїв необхідна наявність таких засобів як блокування доступу до внутрішнього монтажу, до органів управління і комутації пристроїв мають замок дверима, кришками, захисними екранами і т.д.

2.6 Підходи до вирішення задачі аналізу безпеки даних

Локальна комп'ютерна мережа нині не може експлуатуватися автономно, без взаємодії з іншими мережами. Будь-яка організація, чи то приватне підприємство, чи орган державного управління, прагне бути представленим в глобальній мережі Internet — власним сайтом, загальнодоступною електронною поштою, доступом співробітників до інформації в глобальній мережі. Саме це і потребує суворо дотримуватися вимог інформаційної безпеки. Взаємовплив деяких мереж може викликати різні загрози для установи. В разі об'єднання комп'ютерних мереж державних установ, підприємств, науково-дослідних інститутів та організацій з глобальною мережею — Інтернет слід очікувати, окрім хуліганських зламів мережі, і кваліфікованого проникнення до корпоративної мережі установи. Тому мережу Інтернет необхідно ізолювати від внутрішньої, де зосереджені узагальнені дані. Для

ізоляції власної комп'ютерної мережі від глобальної використовується ряд способів з метою захисту від перехоплення спеціальними розвідувальними приймачами випромінених комп'ютерами електромагнітних коливань. В мережах, в яких не обертається інформація з обмеженим доступом, для ізоляції, як правило, досить використати фільтруючий маршрутизатор, що виконує роль брандмауера, тобто

шлюзу, який закриває інформаційні ресурси внутрішньої мережі підприємства.

Захист від проникнення з глобальної мережі можна забезпечити лише за допомогою міжмережевих екранів. Найповніша безпека гарантується лише за фізичної ізоляції мережі Інтернет від власної локальної.

Досить часто співробітнику, який працює з інформацією з обмеженим доступом, необхідно увійти в Інтернет. Для забезпечення такої можливості на робочому місці встановлюється два комп'ютера, один з них підключається до локальної мережі підприємства, а другий — до мережі Інтернет. У даному випадку виникають ускладнення від того, що кабелі власної мережі з захистом інформації та кабелі відкритої мережі Інтернет важко розмістити на достатній відстані. Тому інформація, що обертається в локальній мережі, а також усі паразитні випромінення комп'ютерів, наведені на кабелі локальної мережі, можуть наводитись і на кабелі

відкритої мережі Інтернет. Для відкритої мережі характерно, що її прокладають, як правило, неекранованими кабелями і вона являє собою досить довгу антену, що виходить за межі кордонів захищеної території. Саме тому отримати інформацію можна не лише шляхом перехоплення випромінення, а й безпосередньо через підключення до кабелів відкритої мережі.

Аналіз практики свідчить, що успішне застосування методів і засобів захисту від загроз відтоку інформації багато в чому залежить від правильного вибору, в кожному конкретному випадку, адекватних приборів, пристроїв, апаратури, а також відповідності технічних засобів вимогам допустимості. Застосування технічних засобів, як відомо, можливе, якщо гарантується дотримання правомірності, безпеки, нешкідливості, моральноетичних норм суспільства. Оцінка ефективності застосовуваних методів захисту інформації (активного або пасивного) потребує інструментального підтвердження реальних наслідків їх реалізації, тобто, чи унеможливилася розвідка з використанням каналів побічного електромагнітного випромінення комп'ютера та інших радіоелектронних засобів, що розміщені в екранованому приміщені.

Відомо, що інформація зберігається в різному вигляді і передається різними каналами. До основних носіїв інформації належать: ознайомлені особи,

документи, засоби зв’язку і комунікації, апаратура передавання даних, електронні системи опрацювання, зберігання і розподілу інформації. Інформація в системі управління — це дані, що характеризують процеси, які протікають в організації та у зовнішньому середовищі. Інформація, що виникає і використовується всередині мережі суб’єкта господарювання має назву “внутрішня”. Інформація є найважливішим ресурсом в управлінні. В ній знаходять відображення характеристики усіх інших видів ресурсів, а також події, що відбуваються в організації та у зовнішньому середовищі. Ресурси являють собою засоби досягнення мети організації.

Під час вирішення задачі аналізу безпеки даних у розробленому програмному продукті було використано найефективніші інструменти забезпечення безпеки даних та графічне зображення спроб несанкціонованого доступу залежно від часу.

2.7 Розподілена мережева атака типу DDoS

Розподілені мережеві атаки часто називаються розподіленими атаками типу «відмова в обслуговуванні» (Distributed Denial of Service, DDoS). Цей тип атаки використовує певні обмеження пропускної здатності, які характерні для будь-яких мережевих ресурсів, наприклад, інфраструктурі, яка забезпечує умови для роботи сайту компанії. DDoS-атака відправляє на атакується веб-ресурс велика кількість запитів з метою перевищити здатність сайту обробляти їх і викликати відмову в обслуговуванні.

Мережеві ресурси, такі як веб-сервери, мають обмеження за кількістю запитів, які вони можуть обслуговувати одночасно. Крім допустимого навантаження на сервер існують також обмеження пропускної здатності каналу, який з'єднує сервер з Інтернетом. Коли кількість запитів перевищує продуктивність будь-якого компонента інфраструктури, може відбутися наступне:

— істотне уповільнення час відповіді на запити;

— відмова в обслуговуванні всіх запитів користувачів або частини з них.

Як правило, кінцевою метою зловмисника є повне припинення роботи веб-ресурсу, тобто «відмова в обслуговуванні». Зловмисник може також запросити гроші за зупинку такої атаки. У деяких випадках DDoS-атака може бути спробою дискредитувати або зруйнувати бізнес конкурента. DDoS-атака — комплекс дій, здатний повністю або частково вивести з ладу інтернет-ресурс. В якості жертви може виступати практично будь-який інтернет-ресурс, наприклад веб-сайт, ігровий сервер або державний ресурс. На даний момент практично неможлива ситуація, коли хакер поодинці організовує DDoS-атаку. У більшості випадків зловмисник використовує мережу з комп'ютерів, заражених вірусом. Вірус дозволяє отримувати необхідний і достатній віддалений доступ до зараженого комп'ютера. Мережа з таких комп'ютерів називається ботнет. Як правило, в ботнетах присутній координуючий сервер. Вирішивши реалізувати атаку, зловмисник відправляє команду координирующему сервера, який в свою чергу дає сигнал кожному боту розпочати виконання шкідливих мережевих запитів.

Причинами для здійснення DDoS-атак можуть бути найрізноманітнішими: від конкуренції до особистої неприязні. Основні причини DdoS-атак:

— особиста неприязнь;

— для розваги;

— політичний протест (хактивізм);

— конкуренція;

— вимагання або шантаж.

Наслідки DDoS-атак можуть бути найрізноманітнішими, від відключення датацентру сервера до повної втрати репутації ресурсу і клієнтопотоку. Багато організацій з метою економії неусвідомлено вибирають недобросовісних провайдерів захисту, що часто не приносить ніякої користі. Щоб уникнути

подібних проблем рекомендується звертатися до професіоналів у своїй галузі. Одним із найпоширеніших методів нападу є насичення атакованого комп'ютера або мережевого устаткування великою кількістю зовнішніх запитів, таким чином атаковане устаткування не може відповісти користувачам, або відповідає настільки повільно, що стає фактично недоступним. Взагалі відмова сервісу здійснюється примусом атакованого устаткування до зупинки роботи програмного забезпечення/устаткування або до витрат наявних ресурсів, внаслідок чого устаткування не може продовжувати роботу (рисунок 2.2).

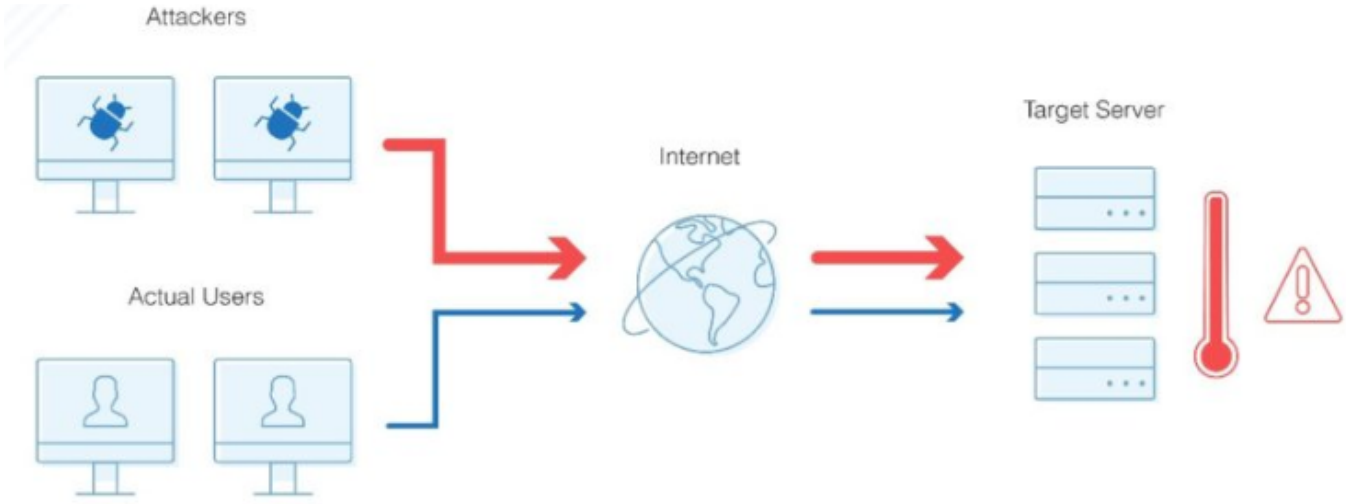

Рисунок 2.2 — Схематичне зображення дії DdoS-атаки

2.8 Висновки до розділу 2

У другому розділі було розглянуто проблеми, які спричинили необхідність розроблення системи підтримки прийняття рішень при аналізі інформаційного забезпечення безпеки даних процесів та систем. Також був проведений огляд існуючих систем для рішення даної проблеми, наведені переваги та недоліки таких систем.

У розділі описано функціональні можливості системи підтримки прийняття рішень при аналізі інформаційного забезпечення безпеки даних

процесів та систем в залежності від потреб ринку інформаційних технологій. Виділений основний функціонал, охарактеризовано загрози, які аналізує розроблена система.

Було надано опис інструментам для забезпечення безпеки даних, їхні переваги та недоліки, також різновиди інструментів та способи їх застосування.

3. МЕТОДИ РЕАЛІЗАЦІЇ СИСТЕМИ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ ПРИ АНАЛІЗІ ІНФОРМАЦІЙНОГО ЗАБЕЗПЕЧЕННЯ БЕЗПЕКИ ДАНИХ

Щоб створити програмне забезпечення для підтримки прийняття рішень при аналізі інформаційного забезпечення безпеки даних, необхідно дослідити технології, які можуть бути використані при розробці такого програмного продукту. Дослідження існуючих технологій є одним із найважливіших факторів на ранньому етапі створення системи, оскільки існує велика кількість інструментів, якими можна буде скористатися для створення кінцевої версії програмного продутку. Вдало обрані технології спрощують початок розробки програмного продукту.

3.1 Середовище розробки Visual Studio 2019

Microsoft Visual Studio — лінія продуктів компанії Microsoft, що включають інтегроване середовище розробки програмного забезпечення і ряд

інших інструментальних засобів. Visual Studio включає в себе редактор вихідного коду з підтримкою технології IntelliSense і можливістю найпростішого рефакторінга коду. Вбудований відладчик може працювати як відладчик рівня вихідного коду, так і відладчик машинного рівня [15]. Решта вбудованих інструментів включають в себе редактор форм для спрощення створення графічного інтерфейсу додатку, веб-редактор, дизайнер класів і дизайнер схеми бази даних. Інтегроване середовище розробки Visual Studio — це оригінальна середа запуску, яка дозволяє редагувати, налагоджувати і створювати код, а потім публікувати додатки. Інтегроване середовище розробки (IDE) — це багатофункціональна програма, яку можна використовувати для різних аспектів розробки програмного забезпечення.

Крім стандартного редактора і відладчика, які існують в більшості середовищ IDE, Visual Studio включає в себе компілятори, засоби виконання коду, графічні конструктори і багато інших функцій для спрощення процесу розробки програмного забезпечення (рисунок 3.1).

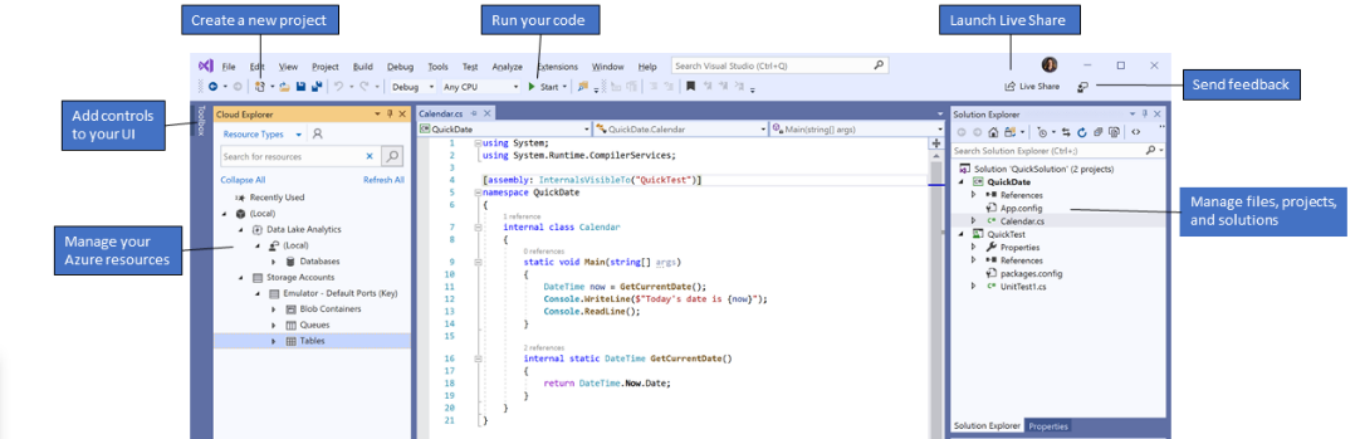

Рисунок 3.1 — Загальний вигляд середовища розробки Visual Studio

Visual Studio також дозволяє створювати і підключати сторонні додатки (плагіни) для розширення функціональності практично на кожному рівні, включаючи додавання підтримки систем контролю версій вихідного коду (Subversion і VisualSourceSafe), додавання нових наборів інструментів (для редагування і візуального проектування коду на предметно-орієнтованих мовах

програмування або інструментів для інших аспектів процесу розробки програмного забезпечення).

Інтегроване середовище розробки Visual Studio пропонує ряд високорівневих функціональних можливостей, які виходять за рамки базового управління кодом. Основні переваги IDE-середовища Visual Studio:

— вбудований Web-сервер. Для обслуговування Web-додатків необхідний Web-сервер, який буде очікувати Web-запити і обробляти відповідні сторінки. Наявність в Visual Studio інтегрованого Web-сервера дозволяє запускати сайт прямо з середовища проектування, а також підвищує безпеку, виключаючи ймовірність отримання доступу до тестового сайту з якого-небудь зовнішнього комп'ютера, оскільки тестовий сервер може приймати з'єднання лише з локального комп'ютера;

— підтримка безлічі мов при розробці. Visual Studio дозволяє писати код своєю рідною мовою чи будь-якою іншою бажаною мовою, використовуючи весь час один і той же інтерфейс. Більш того, Visual Studio також ще дозволяє створювати Web-сторінки на різних мовах, але поміщати їх все в один і той же Web-додаток. Єдиним обмеженням є те, що в кожної Web-сторінці можна використовувати тільки якийсь один мову (очевидно, що в іншому випадку проблем при компіляції було б просто не уникнути);

— менше коду для написання. Для створення більшості додатків потрібно пристойну кількість стандартного стереотипного коду, і Web-сторінки тому не виключення. Наприклад, додавання Web-елемента управління, приєднання обробників подій і коригування форматування вимагає установки в розмітці сторінки ряду деталей. У Visual Studio такі деталі встановлюються автоматично;

— інтуїтивний стиль кодування. За замовчуванням Visual Studio форматує код у міру його введення, автоматично вставляючи необхідні відступи і застосовуючи кольорове кодування для виділення елементів типу коментарів. Такі незначні відмінності роблять код більш зручним для читання і менш схильним до помилок. Застосовувані у Visual Studio автоматичні параметри

форматування можна навіть налаштовувати, що дуже зручно у випадках, коли розробник вважає за краще інший стиль розміщення дужок;

— більш висока швидкість розробки. Багато з функціональних можливостей Visual Studio спрямовані на те, щоб допомагати розробнику робити свою роботу якомога швидше. Зручні функції, на зразок функції IntelliSense (яка вміє перехоплювати помилки і пропонувати правильні варіанти), функції пошуку і заміни (яка дозволяє відшукувати ключові слова як в одному файлі, так і в усьому проекті) і функції автоматичного додавання і видалення коментарів (яка може тимчасово приховувати блоки коду), дозволяють розробнику працювати швидко і ефективно;

— можливості налагодження. Пропоновані в Visual Studio інструменти налагодження є найкращим засобом для відстеження помилок і діагностування поведінки. Розробник може виконувати свій код по рядку за раз, встановлювати точки переривання, при бажанні зберігаючи їх для використання в майбутньому, і в будь-який час переглядати поточну інформацію з пам'яті.

Visual Studio також має і безліч інших функцій: можливість управління проектом; вбудована функція управління вихідним кодом; можливість рефакторізаціі коду; потужна модель розширюваності. Більш того, в разі використання Visual Studio розробник отримує розширені можливості для модульного тестування, спільної роботи і управління версіями коду (що значно більше того, що пропонується в більш простих інструментах).

Microsoft Visual Studio на сьогоднішній день є одним з кращих засобів розробки додатків. З кожною новою версією це середовище набуває все більше і більше корисних функцій.

При написанні коду, його потрібно запустити і перевірити на наявність помилок. Система налагодження Visual Studio дозволяє переглядати код рядок за рядком і одночасно перевіряти змінні. Також є можливість встановити точки перерви, які зупиняють виконання коду на певному рядку. Можна спостерігати за тим, як значення змінної змінюється під час запуску коду (рисунок 3.2).

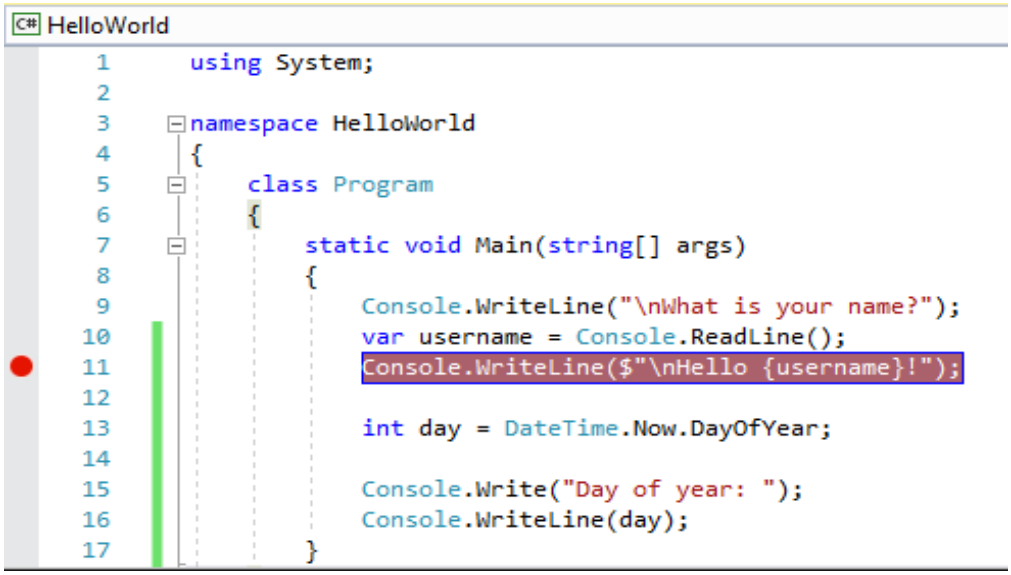

Рисунок 3.2 — Перевірка коду на наявність помилок

У головному вікні після створення нового проекту можна використовувати провідник рішень для перегляду та управління проектом та рішенням та пов'язаними з ними елементами. Solution Explorer у середовищі розробки Visual Studio має дуже зрозумілий інтерфейс та є зручним у використанні (рисунок 3.3).

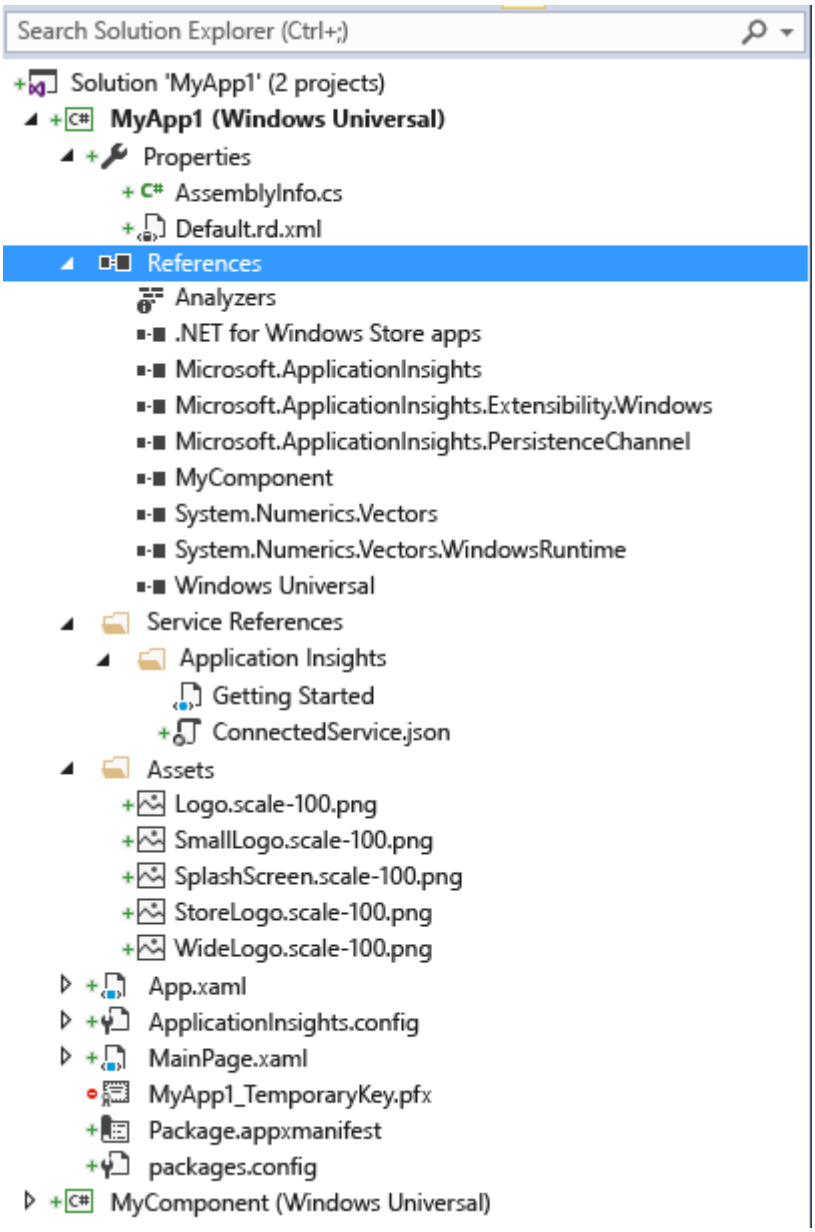

Рисунок 3.3 — Провідник рішень із двома відкритими проектами

3.2 Інструмент для візуального проектування баз даних MySQL Workbench

MySQL Workbench — це уніфікований візуальний інструмент для архітектури баз даних та розробників. MySQL Workbench забезпечує моделювання даних, розробку баз даних та комплексні інструменти адміністрування для налаштування сервера, адміністрування користувача,

резервного копіювання та багато іншого. MySQL Workbench доступний на Windows, Linux та Mac OS X.

MySQL Workbench спрощує розробку та обслуговування бази даних, автоматизує трудомісткі та схильні до помилок завдання, а також покращує обмін даними між командами розробників. Це дозволяє проектувальникам даних візуалізувати вимоги, спілкуватися із зацікавленими сторонами та вирішувати питання дизайну до того, як будуть витрачені великі об'єми часу та ресурсів.

Утиліти для перевірки моделей та схем використовують стандарти найкращої практики моделювання даних, а також застосовують специфічні для MySQL стандарти проектування, щоб не було помилок під час створення нових діаграм або створення фізичних баз даних MySQL.

Для роботи з серверами MySQL в базовий комплект встановлений такий інструмент, як MySQL Workbench. Він представляє графічний клієнт для роботи із серверами, за допомогою якого ми можемо створити, редагувати, видалити базу даних та керувати ними. Так, у Windows після встановлення у меню Пуск ми зможемо знайти програму та запустити її (рисунок 3.4).

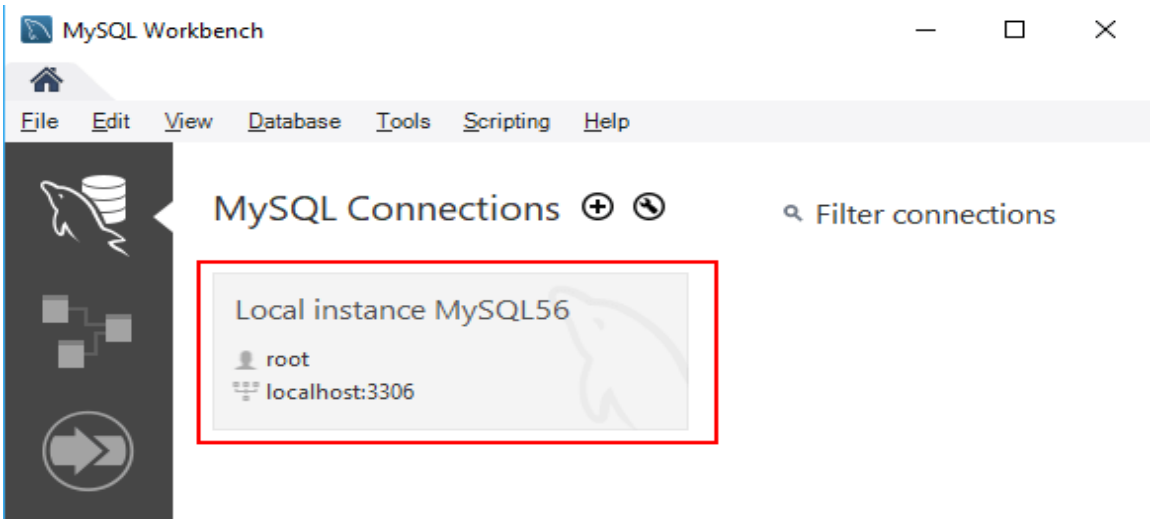

Рисунок 3.4 — Інтерфейс програми MySQL Workbench

Запуститься наступне вікно, де ми можемо побачити поле з назвою

запущеного локального екземпляра MySQL. Натиснувши на нього, відкривається вікно для введення пароля. Тут треба ввести пароль, який був встановлений для користувача root при установці MySQL. Після успішного логіна нам відкриється вміст сервера (рисунок 3.5).

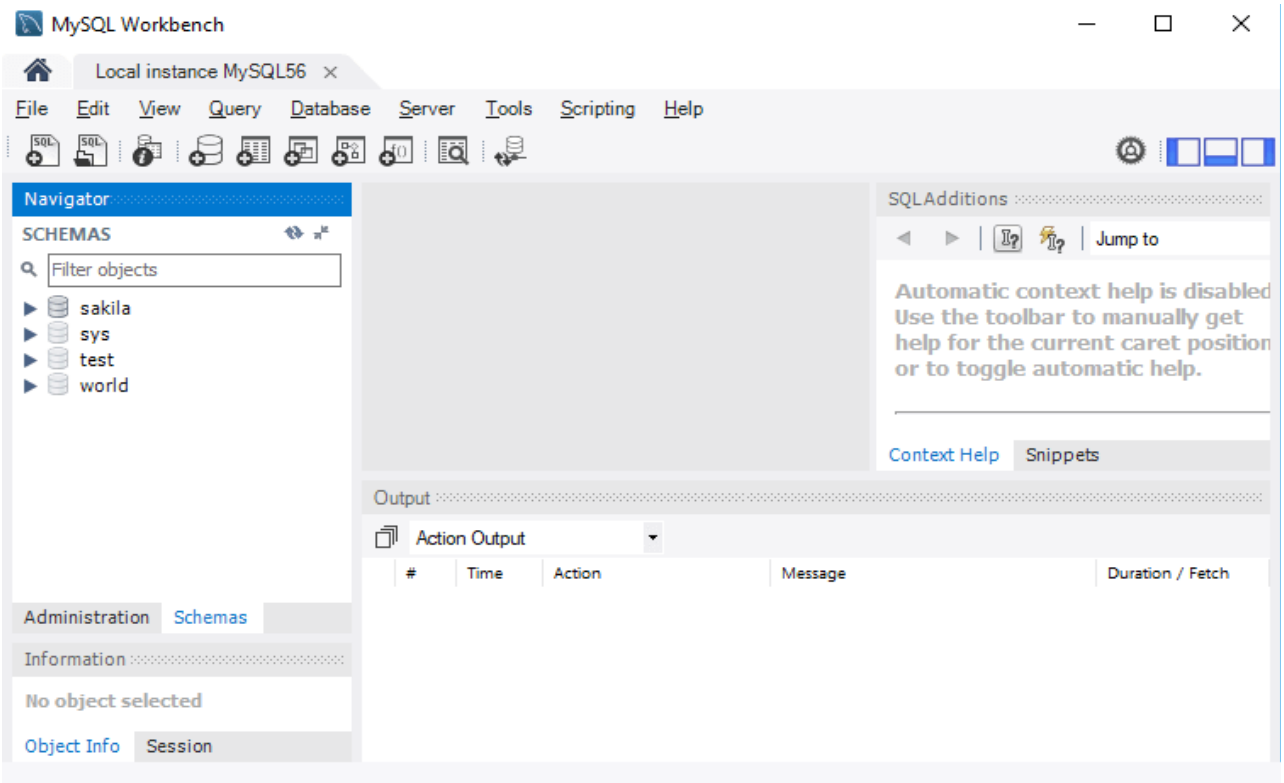

Рисунок 3.5 — Вікно із доступними базами даних

Ми можемо виконувати в цій програмі запити до баз даних. Спочатку необхідно створити саму базу даних. Для цього потрібно натиснути над списком баз даних на значок "SQL" з плюсом. Зокрема, в лівій частині у вікні SCHEMAS можна побачити доступні бази даних. Після цього внизу програми в поле виведення в разі вдалого виконання ми побачимо зелений маркер і звіт про виконання. Таким чином, база даних була створена. Тепер додамо в нею таблицю і дані. Для цього змінюємо код в поле вводу скрипта.

Тобто, ми бачимо що MySQL Workbench є інструментом для візуального проектування баз даних, що інтегрує проектування, моделювання, створення та експлуатацію БД в єдине оточення для системи баз даних MySQL.

Можливості програми:

— дозволяє уявити модель бази даних в графічному вигляді;

— функціональний механізм установки зв'язків між таблицями, в тому числі «багато до багатьох» зі створенням таблиці зв'язків;

— відновлення структури таблиць з уже існуючою на сервері БД;

— зручний редактор запитів, що дозволяє відразу ж відправляти їх із сервером і отримати відповідь у вигляді таблиці;

— можливість редагування даних в таблиці у візуальному режимі.

На ринку існує низка систем управління реляційними базами даних. Приклади реляційних баз даних включають Microsoft SQL Server, Microsoft Access, Oracle, DB2 і т.д. Постає питання, чому для проєктування системи було обрано MySQL над іншими системами управління базами даних. Відповідь на це питання залежить від ряду факторів. MySQL підтримує декілька способів зберігання даних, кожен з яких має свої специфікації, тоді як інші системи, такі як SQL-сервер, підтримують лише один механізм зберігання даних. Його спосіб зберігання за замовчуванням, що постачається з MySQL версії 5.5. підтримує зовнішні ключі для референтної цілісності, а також підтримує транзакції, що відповідають стандарту. Це механізм зберігання даних за замовчуванням для MySQL. Його переваги включають простоту та високу продуктивність. MySQL має високу продуктивність порівняно з іншими системами баз даних. Це пов’язано з її простотою в дизайні та підтримці рішень з декількома накопичувачами. Економічно вигідний, порівняно з іншими реляційними базами даних. MySQLWorkbench це інструмент для проектування та моделювання візуальних баз даних для реляційної бази даних сервера MySQL. Це полегшує створення нових фізичних моделей даних та модифікацію існуючих баз даних MySQL з підтримкою та управлінням змінами.

Адміністрування сервера відіграє вирішальну роль у забезпеченні даних. Основними проблемами, що стосуються адміністрування сервера, є управління користувачами, конфігурація сервера, журнали сервера та багато іншого.

Workbench MySQL має такі функції, які спрощують процес адміністрування сервера MySQL:

— адміністрація користувачів. Візуальна утиліта для керування користувачами, яка дозволяє адміністраторам баз даних легко додавати нових та видаляти існуючих користувачів, якщо виникає потреба, надавати та скидати привілеї та переглядати профілі користувачів;

— конфігурація сервера. Дозволяє розширити конфігурацію сервера і налаштунки для досягнення оптимальної продуктивності;

— резервне копіювання та відновлення баз даних. Візуальний інструмент для експорту / імпорту файлів MySQL. Файли MySQL містять сценарії SQL для створення баз даних, таблиць, представлень даних, збережених процедур та вставки даних;

— журнали сервера. Візуальний інструмент для перегляду журналів сервера MySQL. Журнали включають журнали помилок, двійкові журнали та журнали InnodDB. Ці журнали стануть у пригоді під час діагностики на сервері.

MySQL — це реляційна база даних з відкритим кодом, яка є кросплатформою.

MySQL підтримує кілька механізмів зберігання даних, що значно покращує налаштування та гнучкість роботи сервера. До версії 5.5 двигуном зберігання даних за замовчуванням був MyISAM, якому не вистачало підтримки транзакцій (версії 5.5); двигуном зберігання за замовчуванням є InnoDB, який підтримує транзакції та зовнішні ключі.

Сервер MySQL може адмініструватися за допомогою ряду інструментів mysql для доступу до сервера, які включають як комерційні, так і продукти з відкритим кодом. Популярні приклади включають:

— phpMyAdmin — інструмент доступу до сервера з відкритим вихідним кодом на веб-платформах;

— SQLYog — орієнтований на платформу Windows, настільний інструмент доступу до комерційного сервера;

— MySQL workbench — інструмент доступу до сервера з відкритим вихідним кодом на платформі.

3.3 Система управління базами даних MySQL

MySQL — вільна реляційна система управління базами даних. Розробку та підтримку MySQL здійснює корпорація Oracle, отримавши права на торговельну марку разом із Sun Microsystems. Продукт розповсюджується як під загальною ліцензією, так і під власною комерційною ліцензією. Крім цього, розробники створюють функціональність за заявою ліцензованих користувачів.

MySQL є рішенням для малих та середніх систем. Входить в сервер WAMP, AppServ, LAMP і в портативні колекції серверів Денвер, XAMPP, VertrigoServ. MySQL, як правило, використовується в якості сервісів, які підтримують локальні або віддалені клієнти, однако входить бібліотека внутрішнього сервера, підтримуючи включення MySQL в автономні програми.

Гнучкість СКБД MySQL забезпечує підтримку більшої кількості типових таблиць: користувачі можуть вибрати як таблиці MyISAM, підтримуючи полнотекстовий пошук, так і таблицю InnoDB, підтримуючи транзакції на рівні окремих записів. Крім того, СКБД MySQL розміщується з спеціальною таблицею пам'яті, демонструючи принципи створення нових типових таблиць. Завдяки відкритій архітектурі в СКБД MySQL постійно з'являються нові типи таблиць.

Це система управління реляційними базами даних, у якій дані зберігаються в окремих таблицях, завдяки чому досягається виграш у швидкості й гнучкості. Таблиці зв'язуються між собою за допомогою відносин, завдяки чому забезпечується можливість поєднувати при виконанні запиту дані з декількох таблиць. SQL як частина системи MySQL можна охарактеризувати як мова структурованих запитів, що використовується для доступу до баз даних.

MySQL це програмне забезпечення з відкритим кодом. Застосовувати його

і модифікувати може будь-хто. Таке ПЗ можна отримувати за допомогою Internet і використовувати безкоштовно. При цьому кожен користувач може вивчити вихідний код і змінити його у відповідності зі своїми потребами. MySQL складається з двох частин: серверної і клієнтської.

Сервер MySQL постійно працює на комп'ютері. Клієнтські програми (наприклад, скрипти PHP) посилають серверу MySQL SQL-запити через механізм сокетів (тобто за допомогою мережевих засобів), сервер їх обробляє і запам'ятовує результат. Тобто скрипт (клієнт) вказує, яку інформацію він хоче отримати від сервера баз даних. Потім сервер баз даних посилає відповідь (результат) клієнтові (скрипт).

Структура MySQL трирівнева: бази даних - таблиці - записи. Бази даних і таблиці MySQL фізично представляються файлами. Логічно таблиця являє собою сукупність записів. А запису — це сукупність полів різного типу. Ім'я бази даних MySQL унікально в межах системи, а таблиці — в межах бази даних, поля - в межах таблиці. Один сервер MySQL може підтримувати одразу декілька баз даних, доступ до яких може розмежовуватись логіном і паролем.

База даних з точки зору MySQL — це звичайний каталог, який містить файли певного формату. Таблиці складаються із записів, записи складаються з полів. Поле має два атрибути — ім'я і тип. Тип поля може бути:

— цілим числом;

— дійсним;

— рядком;

— бінарним;

— дата і час;

— перерахування і множини.

Основною функцією для з'єднання з сервером MySQL є mysql_connect (), яка підключає скрипт до сервера баз даних MySQL та виконується авторизацію користувача базою даних. Всі параметри даної функції є необов'язковими, оскільки значення за замовчуванням можна прописати у файлі конфігурації

php.ini. Якщо потрібно вказати інші імена MySQL-хоста, користувача і пароль, завжди є можливість це зробити. Параметр $hostname може бути вказаний у вигляді: хост: порт. Функція повертає ідентифікатор (типу int) з'єднання, вся подальша робота здійснюється тільки через цей ідентифікатор. При наступному виконанні функції mysql_connect () з тими ж параметрами нове з'єднання не буде відкрито, а функція поверне ідентифікатор існуючого з'єднання.

Для закриття з'єднання призначена функція mysql_close (int $ connection_id). Взагалі, підключення можна і не закривати — воно буде закрито автоматично по завершенні роботи скрипта. Якщо кількість з'єднань більше одного, вказується ідентифікатор $ connection_id того з'єднання, яке необхідно закрити.

Також є можливість дізнатися значення одного результату запиту. Це можна зробити функцією mysql_result (resourse $ result, int $ row [, mixed $ field]).

Параметр функції задає номер запису, а параметр field — ім'я або порядковий номер поля. Аргументом поля може бути зсув, ім'я поля, або ім'я поля й ім'я таблиці через крапку. Якщо до імені колонки, в запиті, був використаний аліас, використовувати його потрібно замість реального імені колонки.

Працюючи з великими результатами запитів, слід використовувати одну з функцій, які обробляють відразу цілий рядок результату. Так як ці функції повертають значення кількох рядків відразу, вони набагато швидше. Крім того, вказівка чисельного зміщення працює набагато швидше, ніж вказівка колонки, або колонки і таблиці через крапку. Виклики функції mysql_result () не повинні змішуватися з іншими функціями, які працюють з результатом запиту.

Без сумніву можна стверджувати що при розробці веб–проектів, які являють більшість елементів, більшість користувачів для даних різного виду будуть використовувати MySQL. Часто MySQL використовується із PHP.

Багото веб–додатків використовують MySQL в якості компонентів

програмного забезпечення. Популярність використання веб–додатків тісно повязана з популярністю і можливістю його роботи разом з MySQL.

Багато сайтів з високим трафіком використовують MySQL для збереження даних і реєстрації користувачів. Найбільш розпоширений спосіб установки і використання MySQL на платформі Linux, хоча база даних працює і на інших платформах, наприклад Microsoft Windows.

Сервер MySQL і бібліотеки є більшою частиною додатками до ANSI C/ANCI. Для керування базою даних MySQL можна використовувати інструмент командного рядка (команди MySQL і mysqladmin). Користувачі можуть скачати з сайта MySQL інструменти адміністрування GUI (Graphical user interfase– графічний користувацький інтерфейс) MySQL Administrator MySQL Migration Toolkit і MySQL Query Browser. Інструменти GUI обєднані зараз в один пакет під назвою MySQL GUI Tools.

Ще один важливий елемент, це веб–інтерфейс для адміністрування управління базами даних MySQL написаний на мові PHP вільний веб–додаток з відкритим кодом phpMyAdmin. Через браузер користувача можно здійснювати адміністрацію сервера, запускати команди SQL і передивлятись вміст таблиці і даних.

Додаток завоював свою популярність у зв'язку з тим що надав можливість керувати SQL без посередньо введення SQL команд шляхом доступу до інтерфейсу (рисунок 3.6).

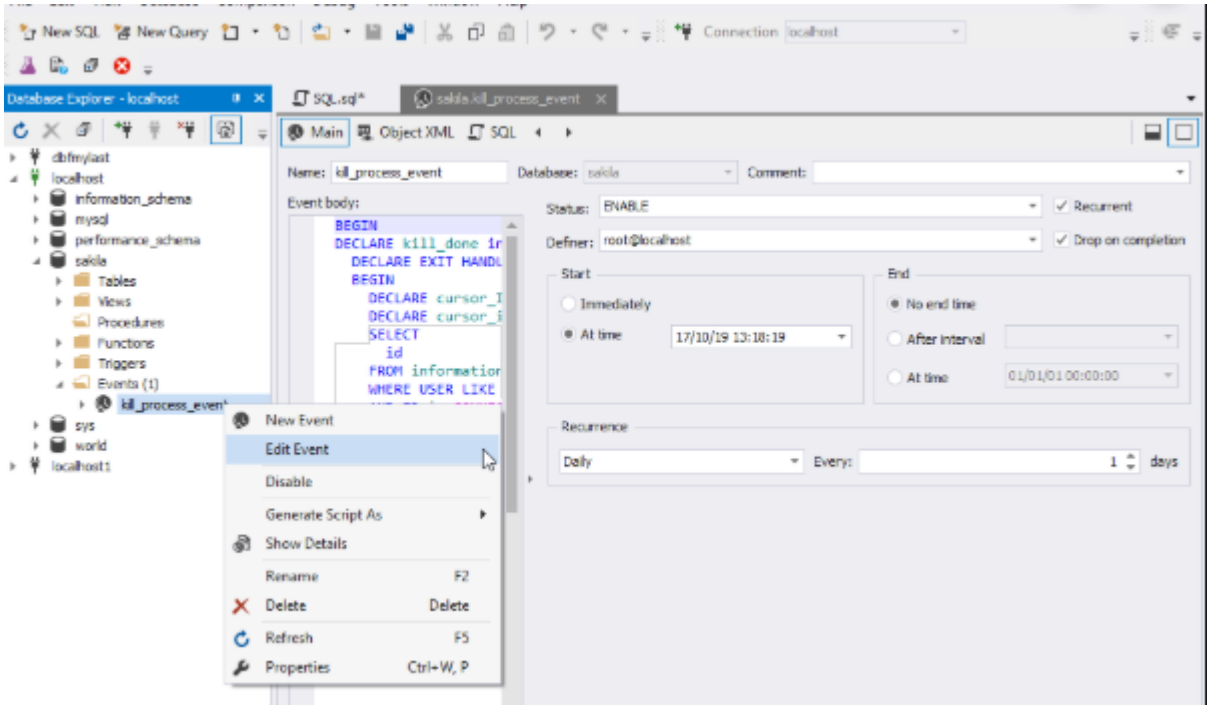

Рисунок 3.6 — Запити у системі управління базами даних

Відкритий вихідний код означає, що можна вільно використовувати і змінювати його. Будь хто може встановити програмне забезпечення. Є можливість вивчити і налаштувати вихідний код, щоб він краще відповідав потребам замовника. Однак GPL (GPU Public License) визначає, що саме ви можете зробити в залежності від умов. Комерційна ліцензована версія доступна, якщо вам потрібно більш гнучке володіння і розширена підтримка. Комп'ютери, які встановлюють і запускають програмне забезпечення СКРБД, називаються клієнтами. Коли їм потрібно отримати доступ до даних, вони підключаються до сервера СКРБД. Це система "клієнт-сервер" (рисунок 3.7).

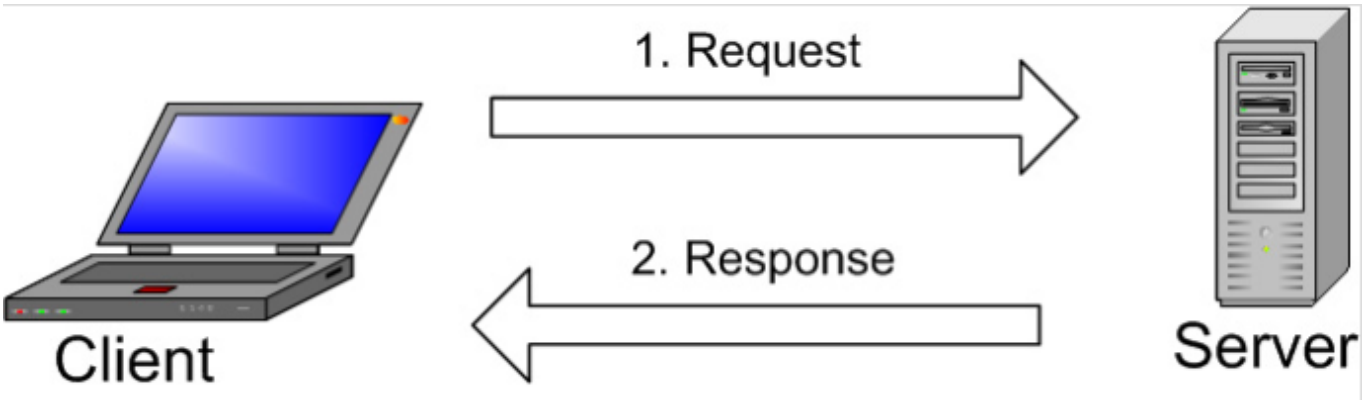

Рисунок 3.7 — Базова структура клієнт-сервер

Зображення пояснює базову структуру клієнт-сервер. Одне або декілька пристроїв (клієнтів) підключаються до сервера через певну мережу. Кожен клієнт може зробити запит з графічного інтерфейсу користувача (GUI) на своїх екранах, і сервер видасть бажаний результат, якщо обидва кінці розуміють інструкцію. Основні процеси, що відбуваються в середовищі MySQL, однакові:

— MySQL створює базу даних для зберігання і управління даними, що визначають відносини кожної таблиці;

— клієнти можуть робити запити, вводячи певні команди SQL на MySQL;

— додаток сервера відповість інформацією і з'явиться на стороні клієнта.

3.4 Об'єктно-орієнтована мова програмування C#

Розроблена система підтримки прийняття рішень при аналізі інформаційного забезпечення безпеки даних була написана мовою програмування C#. Для написання системи було взято до уваги усі переваги даної мови програмування [16].

Мова C# відноситься до родини мов із C-подібним синтаксисом, з них його синтаксис найбільш близький до C ++ і Java. Мова має строгу статичну типізацію, підтримує поліморфізм, перевантаження операторів, вказівники на функції-члени класів, атрибути, події, властивості, винятки, коментарі у форматі XML. Перейнявши багато від своїх попередників — мов C ++, Delphi, — C#, спираючись на практику їх використання, виключає деякі моделі, що зарекомендували себе як проблематичні при розробці програмних систем: так, C# не підтримує множинне успадкування класів (на відміну від C ++) або

виведення типів (на відміну від Haskell).

Мова програмування C# є об'єктно-орієнтованою мовою, але підтримує також і компонентно-орієнтоване програмування. Розробка сучасних додатків все більше тяжіє до створення програмних компонентів у формі автономних пакетів, що реалізують окремі функціональні можливості. Важлива особливість таких компонентів — це модель програмування на основі властивостей, методів і подій.

Кожен компонент має атрибути, які надають декларативні відомості про компоненти, а також вбудовані елементи документації. C# надає мовні конструкції, безпосередньо підтримують таку концепцію роботи. Завдяки цьому C# відмінно підходить для створення і застосування програмних компонентів.

Ось лише кілька функцій мови C#, що забезпечують надійність і стійкість додатків: сбірка “сміття” автоматично звільняє пам'ять, зайняту непотрібними і невикористовуваними об'єктами; обробка виключень надає структурований і розширюваний спосіб виявляти і обробляти помилки; сувора типізація мови не дозволяє звертатися до неініціалізованих змінних, виходити за межі індексованих масивів або виконувати неконтрольоване приведення типів.

У C# існує єдина система типів. Всі типи C#, включаючи типи-примітиви, такі як int і double, успадковують від одного кореневого типу object. Таким чином, всі типи використовують загальний набір операцій, і значення будь-якого типу можна зберігати, передавати і обробляти схожим чином. Крім того, C# підтримує призначені для користувача посилальні типи і типи значень, дозволяючи як динамічно виділяти пам'ять для об'єктів, так і зберігати спрощені структури в стеці.

Щоб забезпечити сумісність програм і бібліотек C# при подальшому розвитку, при розробці C# багато уваги було приділено управлінню версіями. Багато мов програмування обходять це питання, і в результаті програми на цих мовах ламаються частіше, ніж хотілося б, при виході нових версій залежних бібліотек. Питання управління версіями істотно вплинули на такі аспекти

розробки C#, як роздільні модифікатори virtual і override, правила вирішення перевантаження методів і підтримка явного оголошення членів інтерфейсу.

Файли вихідного коду C# зазвичай мають розширення .cs. Код нашої програми Hello World зберігається в файлі hello.cs (рисунок 3.8).

```
using System;

class Program
{
    static void Main(string[] args)
    {
        Console.WriteLine("Hello, World!");
    }
}
```

Рисунок 3.8 — Приклад коду на мові C#

Типова програма починається з директиви using, яка посилається на простір імен System. Простори назв дозволяють ієрархічно впорядковувати програми і бібліотеки C#. Простори назв містять типи і інші простори імен. Наприклад, простір імен System містить кілька типів (в тому числі який використовується в нашій програмі клас Console) і кілька інших просторів імен, таких як IO і Collections.

Директива using, яка посилається на простір імен, дозволяє використовувати типи з цього простору імен без вказівки повного імені. Завдяки директиві using в коді програми можна використовувати скорочене ім'я Console.WriteLine замість повного варіанту System.Console.WriteLine.

Клас, оголошений в програмі, має тільки один член — це метод з ім'ям Main. Метод Main оголошений з модифікатором static. Методи примірника можуть посилатися на конкретний екземпляр об'єкта, використовуючи ключове слово this, а статичні методи працюють без посилання на конкретний об'єкт. За

стандартом точкою входу програми є статичний метод з ім'ям Main.

Вихідні дані програми створюються в методі WriteLine класу Console з простору імен System. Цей клас надається бібліотеками стандартних класів, посилання на які компілятор за замовчуванням додає автоматично.

Структура програми:

— організаційна структура C# грунтується на таких поняттях, як програми, простір імен, типи, члени і збірки;

— вирази створюються з операндів і операторів. Вирази повертають значення;

— використовуються інструкції для опису дій, виконуваних програмою;

— класи є найважливішим типом в мові C#. Об'єкти представляють собою екземпляри класів. Класи створюються описом їх членів;

— структурами є сутності для зберігання даних. Від класів вони відрізняються в першу чергу тим, що є типами значень;

— масивами є структури даних, що містять кілька змінних, доступ до яких здійснюється за який обчислюється індексам;

— інтерфейс визначає контракт, який може бути реалізований класами і структурами. Інтерфейс може містити методи, властивості, події і індексатори. Інтерфейс не надає реалізацію членів, які в ньому визначені. Він лише перераховує члени, які повинні бути визначені в класах або структурах, що реалізують цей інтерфейс;

— тип enum є типом значення з набором іменованих констант;

— тип delegate представляє посилання на методи з конкретним списком параметрів і типом значення, що повертається. Делегати дозволяють використовувати методи як сутності, зберігаючи їх у змінні і передаючи в якості параметрів. Принцип роботи делегатів близький до покажчиків функцій з деяких мов, але на відміну від покажчиків функцій делегати є об'єктно-орієнтованими і строго типізований;

— атрибути дозволяють програмам вказувати додаткові описові дані про

типи, члени та інші сутності.

Мова програмування C# є дуже близьким родичем мови програмування Java. Мова Java була створена компанією Sun Microsystems, коли стрімкий розвиток інтернету поставив задачу розсереджених обчислень. Взявши за початок відому мову C++, Java виключила з неї потенційно непотрібні речі (типу вказівників без контролю виходу за межі). Для розсереджених обчислень була створена концепція віртуальної машини та машинно-незалежного байт-коду, свого роду посередника між вихідним текстом програм і апаратними інструкціями комп'ютера чи іншого інтелектуального пристрою.

Java набуває чималої популярності, і була ліцензована також і компанією Microsoft. Але з плином часу компанія розробників почала винуватити Microsoft, що та при створенні мови Java робить її сумісною виключно з платформою Windows, чим суперечить самій концепції машинно-незалежного середовища виконання і порушує ліцензійну угоду. Microsoft відмовилася піти назустріч вимогам компанії, і тому з'ясування стосунків набуває статусу судового процесу. Суд визнав позицію компанії справедливою, і зобов'язав Microsoft відмовитися від позаліцензійного використання Java. У цій ситуації в Microsoft вирішили, користуючись своєю вагою на ринку, створити свій власний аналог Java — мову, в якій корпорація стане повновладним господарем. Ця новостворена мова отримала назву C#. Вона успадкувала від Java концепції віртуальної машини (середовище .NET), байт-коду і більшої безпеки вихідного коду програм, плюс врахувала досвід використання програм на Java.

Нововведенням C# стала можливість легшої взаємодії, порівняно з мовами-попередниками, з кодом програм, написаних на інших мовах, що є важливим при створенні великих проектів. Якщо програми на різних мовах виконуються на платформі .NET, .NET бере на себе клопіт щодо сумісності програм (тобто типів даних, за кінцевим рахунком).

Мова C# підтримує строго типізовані неявні оголошення змінних з ключовим словом var і неявно типізовані масиви з ключовим словом new [], за яким слідує ініціалізатор колекції. C# також застосовує тип даних Boolean, bool.

Умовні вирази, такі як while та if, потребують висловлювання, що реалізує оператор true або false. Хоч C++ також має тип Boolean, він може бути вільно переведений в цілочисельний та з нього, а вирази, такі як if, вимагають тільки того, щоб тип був конвертований в bool, що дозволяє бути змінній int-типом або вказівником. C# забороняє «ціле значення означає справжній або помилковий підхід» на тій підставі, що примус програмістів використовувати вирази, які повертають точно bool, можуть створювати деякі типи помилок програмування, наприклад if (a = b) (використання присвоювання = замість рівності == , які, хоча і не є помилкою на C або C ++, все одно будуть знайдені компілятором).

C# безпечніший якщо порівнювати його із C++. Єдиними неявними перетвореннями за умовчуванням є ті, які вважаються безпечними, наприклад, розширення цілих чисел. Це застосовується під час компіляції і в деяких випадках, під час виконання. Не трапляється неявних перетворень між булевими і цілими числами, а також між членами перерахування і цілими числами (крім літерала 0, який може бути неявно перетворений в будь-який нумерований тип). Будь-яке призначене для користувача перетворення повинно бути явно позначене як явне або неявне, на відміну від конструкторів копіювання C++ і операторів перетворення, які за умовчанням є неявними. C# має явну підтримку ковариації та контраваріантності в родових типах, на відміну від C++, яка має певний рівень підтримки контраваріантності просто через семантику типів, що повертаються, на віртуальні методи. Члени перерахування розміщуються в своєму власному обсязі.

Мова C # не допускає глобальних змінних або функцій. Всі методи і члени повинні бути оголошені всередині класів. Статичні члени відкритих класів можуть замінювати глобальні змінні та функції.

3.5 Висновки до розділу 3

В розділі були розглянуті основні інструменти для розробки системи

підтримки прийняття рішень при аналізі інформаційного забезпечення безпеки даних.

Для написання програмного продукту було обрано мову програмування C#. У даному розділі було наведено переваги цієї мови, порівняння з аналогами та можливості її функціоналу.

Розробка програмного продукту також включала в себе такі інструменти як MySQL Workbench, MySQL та середовище розробки Visual Studio. Обгрунтування такого вибору було наведено у розділі 3. Для виконання дипломної роботи важливо провести аналіз існуючих інструментів, адже це буде впливати на час виконання, якість, швидкість роботи продукту та її якість.

4. ВЗАЄМОДІЯ З КОРИСТУВАЧЕМ

У цьому розділі буде розглянуто основні можливості розробленого продукту та взаємодія користувача з ним. Також буде переглянуто системні вимоги до комп'ютера. Користувачеві не потрібно встановлювати ніяких додаткових інструментів, програмний модуль являє собою додаток що запускається у вигляді форми.

4.1 Опис бази даних системи

Для зв’язку додатку із базою даних MySQL був використаний MySqlConnector.

MySqlConnector – постачальник даних ADO.NET для MySQL Server, MariaDB, Percona Server, Amazon Aurora, база даних Azure для MySQL, Google Cloud SQL для MySQL тощо. Він забезпечує реалізацію DbConnection, DbCommand, DbDataReader, DbTransaction – класи, необхідні для запиту та оновлення баз даних з керованого коду. Для роботи з MySQL необхідно у коді додати бібліотеку MySQL Connector: using MySql.Data.MySqlClient; для запитів і здійснення маніпуляцій за базами даних необхідно використовувати об’єкти класу MySqlConnection.

Вся взаємодія між .NET-додатком та сервером MySQL здійснюється шляхом об’єкта MySqlConnection при використанні класичного протоколу MySQL. Перш ніж програма може взаємодіяти з сервером, вона повинна створити об'єкт MySqlConnection, створити налаштування та відкрити MySqlConnection.

Для з’єднання з необхідною базою даних необхідно використати такі змінні:

— connection: буде використовуватися для відкриття з'єднання з базою даних;

— server: вказує, де розміщується наш сервер, у нашому випадку - localhost;

— database: це ім'я бази даних, яку ми будемо використовувати;

— uid: це наше ім'я користувача MySQL;

— password: це наш MySQL пароль;

— connectionString: містить рядок з'єднання для підключення до бази даних і буде призначений змінній з'єднання.

Перш ніж починати роботу с MySqlConnector необхідно в проекті додати посилання до нього (рисунок 4.1).

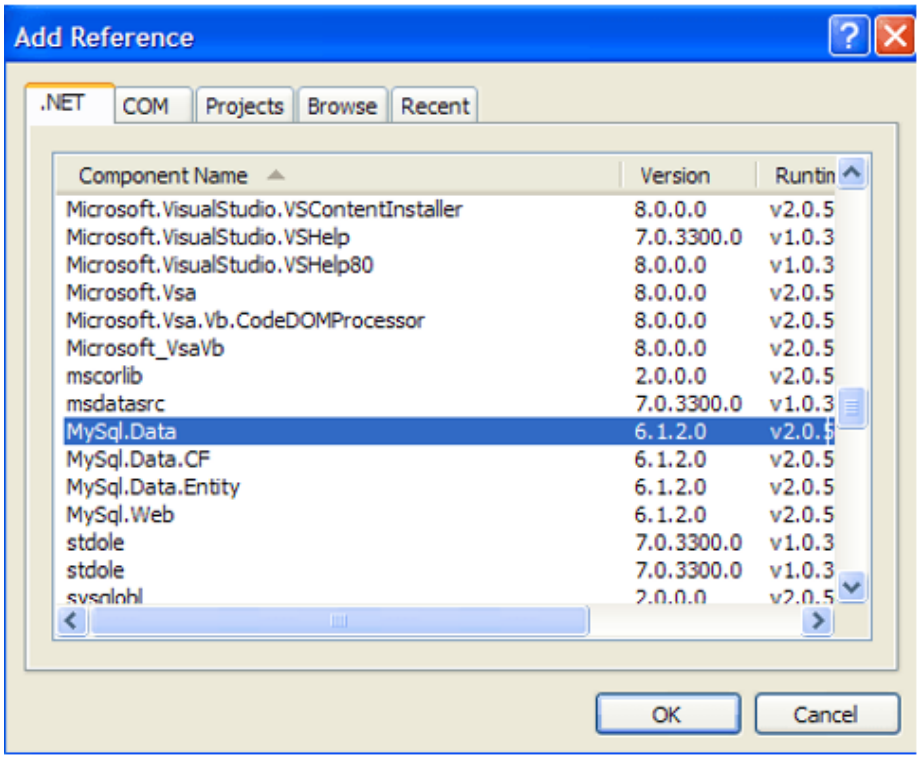

Рисунок 4.1 — Додавання MySQLConnector до проекту

База даних розробленої системи складається з 4 таблиць: «users» (Користувачі), «number_of_attacks» (кількість атак), «db_attacks» (атаки на базу даних), «db_savings» (збереження баз даних).

Таблиця «users» призначена для зберігання даних про користувачів системи. В ній зберігається логін, пароль, ім’я користувача, роль(адміністратор, користувач), посада, організація в якій працює користувач (рисунок 4.2).

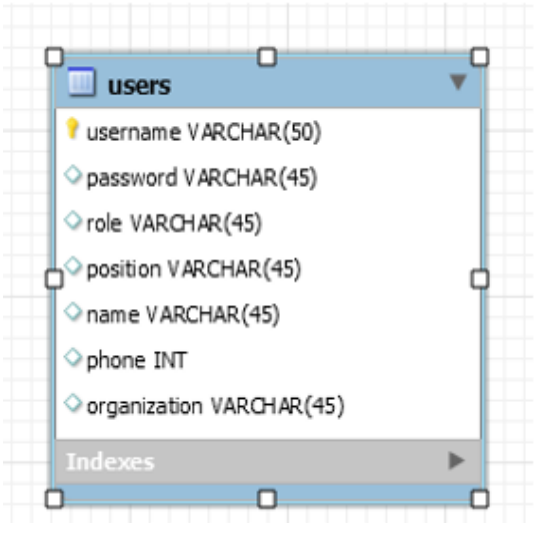

Рисунок 4.2 — Структура таблиці «users»

Таблиця «number_of_attacks» призначена для зберігання даних про атаки на систему. Зокрема, вона зберігає дані про час атаки, кількість атак, і порти на які було здійснено атаки (рисунок 4.3).

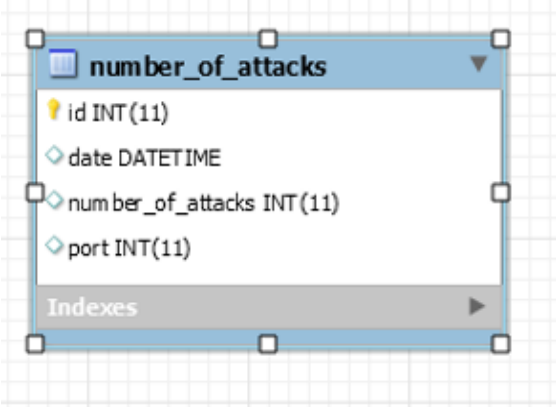

Рисунок 4.3 — Структура таблиці «number_of_attacks»

Таблиця “db_attacks” призначена для зберігання даних про атаки на бази даних. Зокрема, вона зберігає дані про час атак, їх кількість, а також імена баз даних на які була здійснена та чи інша атака (рисунок 4.4).

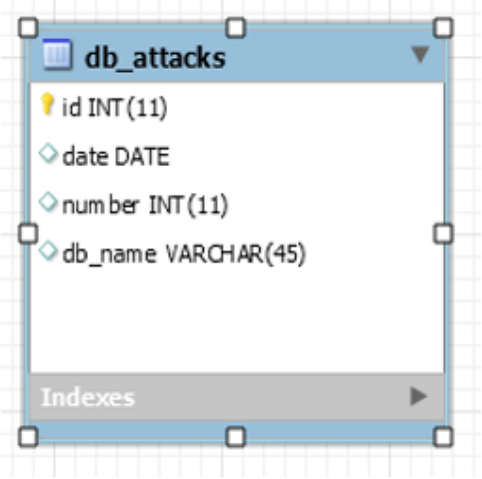

Рисунок 4.4 — Структура таблиці «db_attacks»

Таблиця «db_savings» призначена для зберігання інформації про історію збережень баз даних. Зокрема, про те, яку базу даних було збережно, дату збереження і хто виконував збереження (рисунок 4.5).

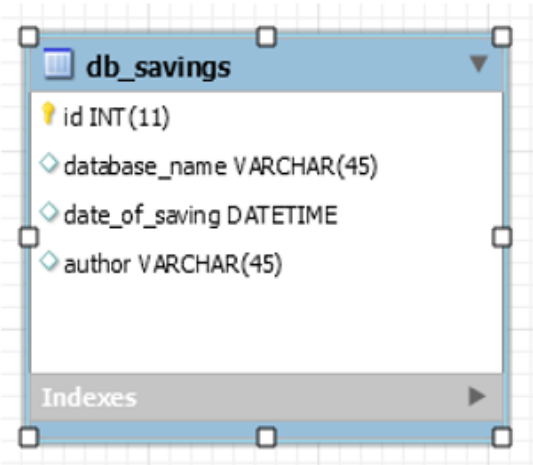

Рисунок 4.5 — Структура таблиці «db_savings»

4.2 Системні вимоги та інсталяція

Розроблений програмний продукт, а саме його клієнтська частина працює на будь-якому персональному комп'ютері під управлінням операційної системи Windows. Рекомендовані вимоги до конфігурації ПК, який виступає в ролі сервера:

— операційна система Microsoft Windows 2000/XP/Server 2003/Vista;

— встановлена СКБД MySQL;

— налаштований вервер Apache;

— оперативна пам'ять 1024 Мб і вище;

— кількість вільного місця на жорсткому диску близько 150 Мбайт;

— роздільна здатність екрану 1280x800 пікселів;

— тактова частота процесору 1200 МГц;

— доступ до мережі Інтернет.

Рекомендовані вимоги до конфігурації пристрою, який виступає в ролі кінцевого споживача:

— оперативна пам'ять 512 Мб;

— тактова частота процесору 800 МГц;

— роздільна здатність екрану 1280x800 пікселів;

— кількість вільного місця на жорсткому диску 30 Мб;

— доступ до мережі Інтернет;

— операційна система Windows.

Для користування програмою на клієнтському комп’ютері потрібно відкрити її за допомогою ярлика програми. Для цього потрібно встановити програму на свій комп’ютер.

На комп’ютері, який виступає сервером, необхідно встановити MySQL Server для управління базою даних.

4.3 Сценарій роботи користувача

Основна розробка програмного продукту поділялася на 2 частини: побудова інтерфейсу та бізнес-логіки програмного продукту та проектування та організація роботи бази даних. Після запуску програмного застосунку користувач потрапляє на головну сторінку, де повинен пройти авторизацію для користування системою (рисунок 4.6).

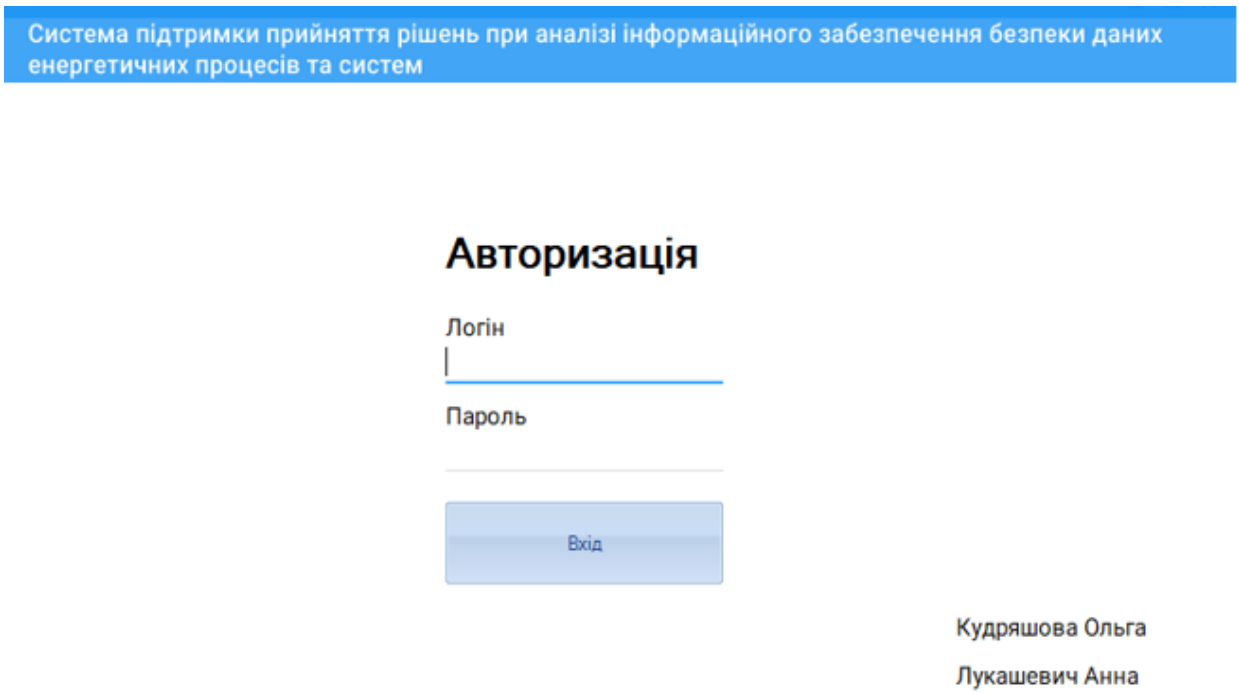

Рисунок 4.6 — Головна сторінка системи

Після входу жо системи для користувача відкривається головне вікно системи, яка складається з двох частин: підсистеми аналізу та підсистеми збереження даних (рисунок 4.7).

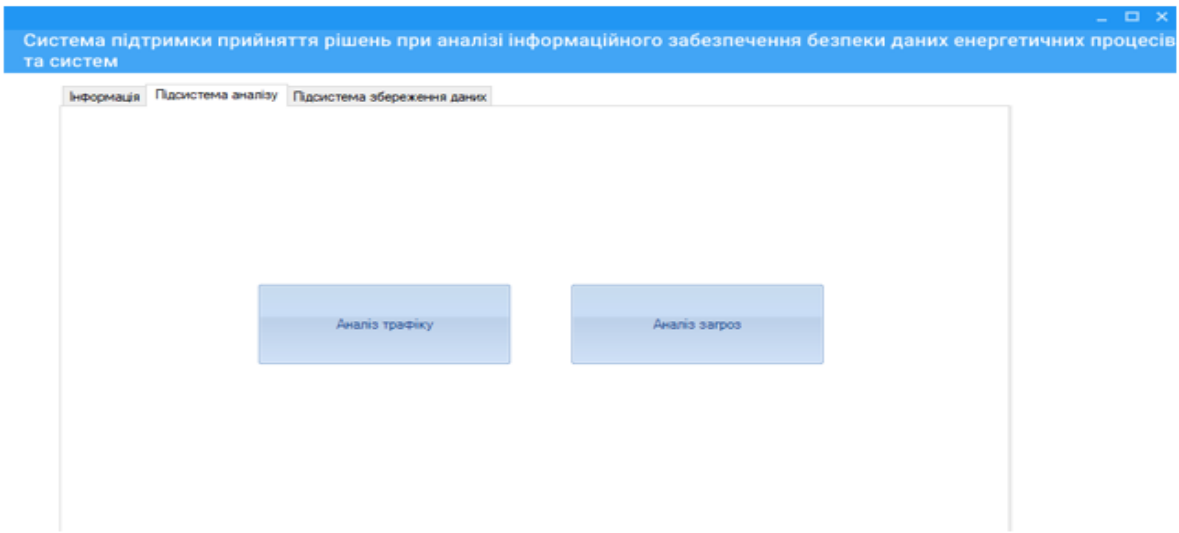

Рисунок 4.7 — Головне вікно системи

Користувач має натиснути вкладку «Підсистема аналізу», щоб перейти до підсистеми аналізу (рисунок 4.8).

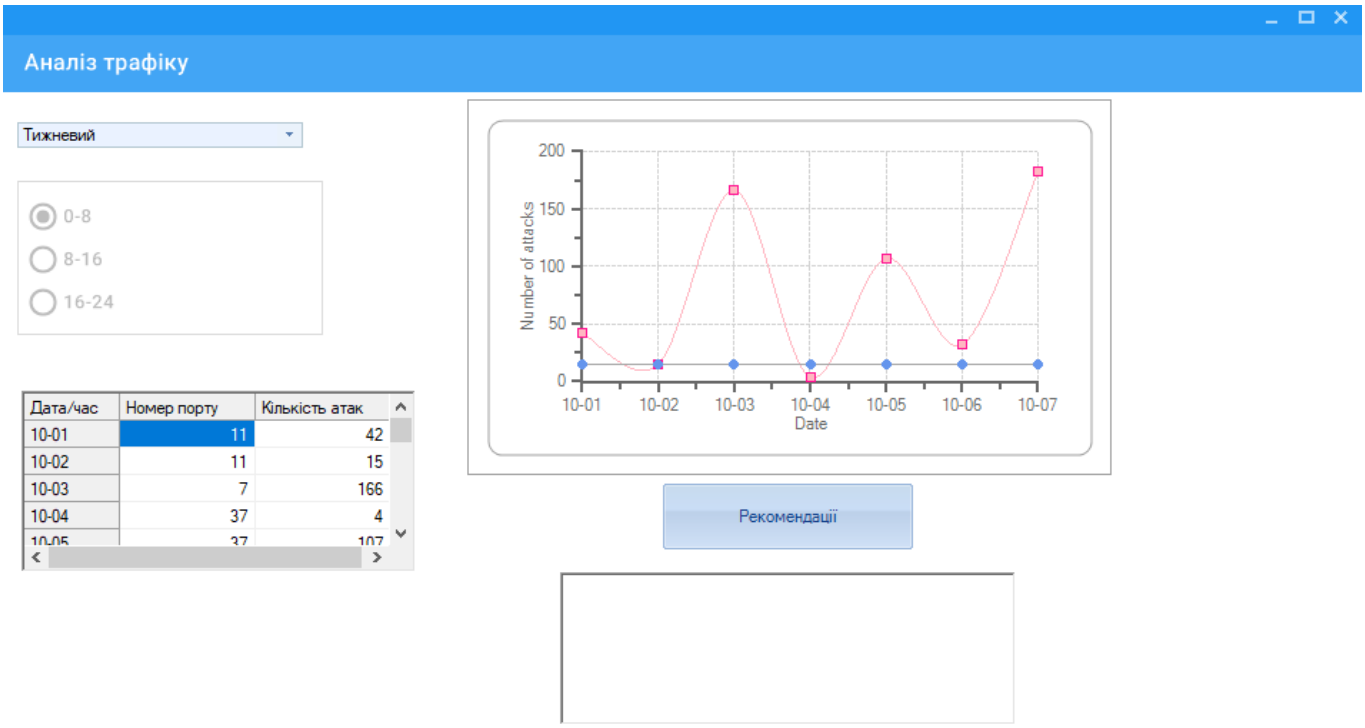

Рисунок 4.8 — Вікно підсистеми аналізу трафіку

У даній підсистемі користувач має змогу побачити два графіка відношень кількості несанкціонованих спроб до часу (рисунок 4.9).

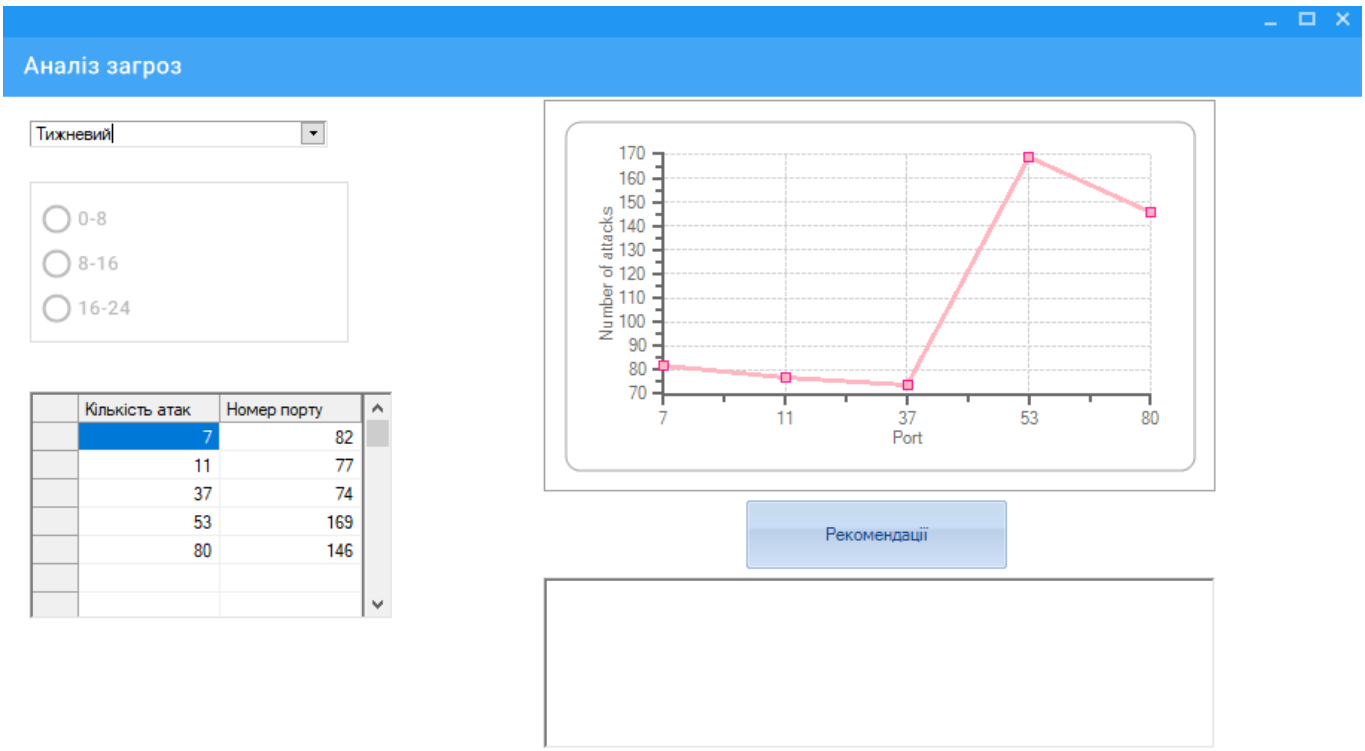

Рисунок 4.9 — Вікно підсистеми аналізу загроз

Крім цього, користувач системи має змогу переглянути графіки, що демонструють спроби атак на систему, їх кількість в залежності від часу. Можливий перегляд графіків за тиждень і за день. Зокрема, можливий перегляд даних і у табличному вигляді, оскільки для деяких користувачів такий спосіб є більш зручним та наочнішим.

4.4 Висновки до розділу 4

У даному розділі було представлено інструкції щодо інсталяції та системних вимог для встановлення додатку, щодо роботи із системою підтримки прийняття рішень при аналізі інформаційного забезпечення безпеки даних енергетичних процесів та систем. Підсистема аналізу.

Продемонстровано можливі сценарії роботи користувача з системою, показано зручність інтрефейсу користувача.

5. СТАРТАП ПРОЕКТ

Розділ має за мету проведення маркетингового аналізу проекту як стартапу для аналізу та власне можливості його ринкового впровадження та можливих способів та напрямів реалізації цього впровадження. Проведення маркетингового аналізу означає виконання певних визначених кроків та завдань.

5.1 Опис ідеї проекту

В даному підрозділі слід проаналізувати та подати у вигляді таблиць:

— зміст ідеї (що пропонується);

— можливі напрямки застосування;

— основні вигоди, що може отримати користувач товару;

— чим відрізняється від існуючих аналогів та замінників.

Перші три пункти подаються у вигляді таблиці (таблиця 5.1) і дають цілісне уявлення про зміст ідеї та можливі базові потенційні ринки.

Таблиця 5.1 — Опис ідеї стартап-проекту

Зміст ідеї	Напрямки застосування	Сегменти споживачів
Розробка підсистеми аналізу	1. Аналіз можливих атак на систему	Державні структури та компанії
	2. Видача рекомендацій користувачу щодо захисту системи	Приватні фірми, банківська сфера

Аналіз потенційних технолгічно-економічних переваг ідеї (різниця у порівнянні з конкуруючими приграмними рішеннями) порівняно із пропозиціями конкурентів передбачає:

— аналіз та ідентификація всіх технологічно-економічних властивостей та характеристик ідеї;

— проведення дослідження та визначення потенційного кола конкурентів (програм-конкурентів) або товарів-замінників чи товарів-аналогів, що наразі існують на ринку, та проведення збору інформації та даних щодо значень технологічно-еконо¬мічних показників щодо власного проекту та програм-конку-рентів відповідно до зазначеного переліку;

— проведення порівняльного аналізу показників: для власної ідеї визначаються показники, що мають а) гірші значення (W, слабкі); б) аналогічні (N, нейтральні) значення; в) кращі значення (S, сильні) (таблиця 5.2).

Таблиця 5.2 — Визначення сильних, слабких та нейтральних характеристик

No п/п		**Потенційні програми/концепції конкурентів**		
		Мій проект	Система	Система
1	W слабка сторона	Відносно повільний час роботи	Повільний час роботи	Не є направленими на конкретну предметну область.
2	N нейтральна сторона	Можливіть роботи без доступу до інтернет	Занадто комплексна для простого користувача	Надто багато функцій
3	N нейтральна сторона	Невелика ресурсозатратність при роботі користувача	Потребує великої кількості ресурсів комп'тера для виконання	Потребує великої кількості ресурсів комп'тера для виконання

Таблиця 5.2 (Продовження)

4	S сильна сторона	Можливість засосовувати в установах різного спрямування	Можливість засосовувати в установах різного спрямування	Можливість засосовувати в установах різного спрямування
5	S сильна сторона	Своєчасне оновлення	Ширший функціонал	Ширший функціонал

5.2 Технологічний аудит ідеї проекту

В цьому підрозділі зроблено аналіз технологій, за допомогою якоих можна реалізувати ідею проекту. Визначення технологічної здійсненності ідеї проекту передбачає аналіз таких складових (таблиця 5.3):

— за якою технологією буде виготовлено продукт згідно з ідеєю проекту;

— чи існують такі технології, чи їх потрібно розробити/доробити;

— чи доступні такі технології авторам проекту;

Таблиця 5.3 – Технологічна здійсненність ідеї проекту

No п/п	Ідея проекту	Технології і реалізації	Наявність технологій	Доступність технологій
1	Інтерфейс користувача	Мова програмування C#	Наявна	Умовна безкоштовно
2	База даних	СКБД MySQL	Наявна	Умовна безкоштовно

Таблиця 5.3 (продовження)

3	Алгоритм створення звітів	Мова програмування C#	Відсутня	Відсутня
4	Алторитм формування сценарію	Мова програмування C#	Відсутня	Відсутня
Висновок: проект реалізувати можливо				

За результатами аналізу таблиці можна зробити висновок щодо можливості технологічної реалізації проекту: так чи ні, а також технологічного шляху, яким цього варто досягти (з поміж названих технологій обираються такі, що доступні авторам проекту та є наявними на ринку).

5.3 Аналіз ринкових можливостей запуску стартапу

Передбачання і визначення ринкових можливостей, які можна використати під час ринкової імплементації проекту, та ринкових загроз, які можуть перешкодити реалізації проекту, дає змогу спланувати напрями розвитку проекту із урахуванням стану ринку, аналгічних пропозицій проектів-конкурентів.

Спочатку проводиться аналіз попиту: наявність попиту, обсяг, динаміка розвитку ринку (таблиця 5.4).

Таблиця 5.4. — Характеристика потенційного ринку стартап-проекту

No п/п	Показники стану ринку (найменування)	Характеристика
1	Кількість головних гравців, од	3

2	Загальний обсяг продажів, грн/ум.од	250
3	Динаміка ринку (якісна оцінка)	Зростає

Таблиця 5.4 (продовження)

4	Наявність обмежень для входу (вказати характер обмежень)	Немає
5	Специфічні вимоги до стандартизації та сертифікації	Немає
6	Середня норма рентабельності в галузі (або по ринку), %	50

Середня норма рентабельності в галузі (або по ринку) порівнюється із банківським відсотком на вкладення. І якщо останній є вищим, можливо, є сенс вкласти кошти в інший проект.

За результатами аналізу таблиці робиться висновок щодо того, чи є ринок привабливим для входження за попередньою оцінкою.

Надалі визначаються потенційні групи клієнтів, їх характеристики, та формується орієнтовний перелік вимог до товару для кожної групи (таблиця 5.5).

Таблиця 5.5 — Характеристика потенційних клієнтів стартап-проекту

No п/п	Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
1	Надання підсистеми збереження даних, яку потребує ринок компаній, для яких важлива	ЗВО, великі корпорації, банківські установи, науковці	Компанії заключають довготривалі договори, а стартапери віддають	стабільність роботи; невисока ціна; наявність випробувального періоду;

	безпека даних		перевагу пробному терміну	наявність документації; підтримка необхідних платформ.

Після визначення потенційних груп клієнтів проводиться аналіз ринкового середовища: складаються таблиці факторів, що сприяють ринковому впровадженню проекту, та факторів, що йому перешкоджають (таблиці 5.6-5.7).

Надалі проводиться аналіз пропозиції: визначаються загальні риси конкуренції на ринку. Аналіз пропозиції необхідно виконати аналізуючи існуючі види конкуренцій.

Необхідною умовою ефективного функціонування механізму саморегулювання ринкової економіки є конкуренція. Вона є важливою рушійною силою розвитку ринкової економічної системи. Конкуренцію породжують об'єктивні умови ринкового господарювання: різні форми власності на засоби виробництва, ринки збуту виробленої продукції, сфери використання капіталу з метою отримання найбільшого прибутку.

Конкуренція — це суперництво (змагальність) між різними учасниками ринкової економіки за найбільш вигідні умови виробництва та реалізації товарів і послуг, за привласнення найбільшого прибутку. Вона виступає силою, яка мобілізує особистий економічний інтерес і підприємницький потенціал та спрямована на їх максимальну реалізацію.

Захист конкуренції, суб'єктів господарювання і споживачів від недобросовісної конкуренції передбачає демонополізацію вітчизняної економіки і створення ринкового конкурентного середовища.

Таблиця 5.6 — Фактори загроз

No п/п	**Фактор**	**Зміст загрози**	**Можлива реакція компанії**
1	Підходить для нових проектів	Потребує визначеної структури бази даних	Імпорт схеми бази даних

Таблиця 5.6 (продовження)

2	Власний формат та вигдляд відображення ДС	При необхідності потрібно розробка сервісу преведення до визначенного формату	Додавання можливості автоматизованого експорту в різні типи сховищ, розробка додаткового ПЗ
3	Обмеженість функцій	Інструмент обмежений наявними функціями і не має деяких функцій, які мають конкуренти	Додавання нових функцій за потреби

Таблиця 5.7 — Фактори можливостей

No п/п	**Фактор**	**Зміст можливості**	**Можлива реакція компанії**
1	Незалежність від платформи	Можна використовувати на Linux, Windows, Mac операційних системах	Вихід на мобільний ринок, вихід на рівень web додатків
2	Недоліки в існуючих альтернативах	Можна використовувати на Linux, Windows, Mac операційних системах	Модифікація існуючих платформ

Фінальним етапом ринкового аналізу можливостей впровадження проекту є складання SWOT-аналізу (матриці аналізу сильних (Strength) та слабких (Weak) сторін, загроз (Troubles) та можливостей (Opportunities) на основі виділених ринкових загроз та можливостей, та сильних і слабких сторін (таблиця 5.8).

Таблиця 5.8 — Складання SWOT-аналізу

Сильні сторони: Актуальність користування системою, яка викликана бажанням забезпечення безпеки даних Оцінка проходить відразу для великої кількості людей, а також у будь-який період часу. Актуальність користування системою, яка викликана постійним розвитком інформаційниз технологій, невелика ціна користування за місяць	Слабкі сторони: Потребує масштабної рекламної компанії Орієнтація на комп'ютерні додатки, які можуть відсіяти «не розвинутих» в технічному плані клієнтів Дороге зберігання великої кількості даних Обробка даних
Можливості: Можливе продовження розробки проекту за кордоном, тому що проблема безпеки даних актуальна не лише в Україні Можливість створення звітів за	Загрози: Відсутність користувачів через погану рекламну компанію Не можливість достукатися до необхідних API Втрата конфіденційних даних.

виконаними аналізами Збереження результатів у різних форматах Зручність у використанні	

5.4 Висновки до розділу 5

Розроблений програмний продукт має переваги над існуючими конкурентами та є конкурентноздатним на ринку. Програма має шляхи подальшого розвитку, визначені маркетингові стратегії та шляхи збуту. Основна цільова аудиторія — підприємтсва, які потребують підвищеного рівня безпеки.

ВИСНОВКИ

Для вирішення проблем, пов’язаних із захистом інформації, спеціалісти по забезпеченню безпеки даних повинні мати стратегію інформаційної безпеки, необхідне програмне забезпечення та відповідні інструменти. Безпека та захист в інформаційних системах мають будуватись з урахуванням комплексного підходу до побудови системи захисту, що передбачає об'єднання в єдиний комплекс необхідних заходів та засобів захисту інформації на всіх рівнях системи інформаційного забезпечення. Саме таким засобом є розроблена система підтримки прийняття рішень при аналізі інформаційного забезпечення безпеки даних, адже головною її метою є відстеження спроб несанкціонованого доступу та аналіз загроз, що теж є дуже важливим фактором із забезпечення безпеки.

Забезпечення інформаційної безпеки даних це забезпечення стану захищеності систем обробки і зберігання даних, при якому забезпечено конфіденційність, доступність і цілісність інформації, або комплекс заходів, спрямованих на забезпечення захищеності інформації від несанкціонованого доступу, використання, оприлюднення, руйнування, внесення змін, ознайомлення, перевірки запису чи знищення. Тому дуже важливим є саме фіксування кількості атак, часу атак та номера портів, на які були спричинені атаки, оскільки ці дані у подальшому можуть бути використані для формування стратегії забезпечення безпеки.

В даній роботі було проаналізовано можливі загрози забезпеченню безпеки даних, спроби несанкціонованого доступу до системи та видача рекомендацій користувачу щодо зменшення кількості атак на систему.

Отже, розроблений програмний продукт може бути застосований у будь-якій сфері життєдіяльності людини, яка потребує підвищенного рівня безпечного збереження даних.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Коваленко, Ю. О. (2010). Забезпечення інформаційної безпеки на підприємстві. Економіка промисловості (3). с. 123–129.

2. Информационная безопасность (2-я книга социально-политического проекта «Актуальные проблемы безопасности социума»). М.: «Оружие и технологии», 2009, с. 213.

3. Аудит інформаційної безпеки: підручник / В. А. Ромака, А. Е. Лагун, Ю. Р. Гарасим та ін. ; Держ. служба України з надзвич. ситуацій, Львів. держ. ун-т безпеки життєдіяльності, НАН України, Ін-т приклад. проблем механіки і математики ім. Я. С. Підстригача. — Львів: Сполом, 2015. — 363 с. : іл. — Бібліогр.: с. 280—281 (37 назв).

4. Інформаційна безпека людини як споживача телекомунікаційних послуг: Монографія / І. В. Арістова, Д. В. Сулацький ; НДІ інформатики і права НАПрН України. — К. : Право України ; Х. : Право, 2013. — 184 с.

5. Кормич Б. А. Інформаційна безпека: організаційно-правові основи: Навч. посібник. — К.: Кондор, 2004. — 384 с.

6. Сідак В. С., Артемов В. Ю. Забезпечення інформаційної безпеки. Навчальний посібник. — К.: КНТ, 2007.

7. Цимбалюк В.С. Проблеми державної інформаційної політики: гармонізація міжнародного і національного інформаційного права // Правове, нормативне та метрологічне забезпечення системи захисту інформації в Україні. — К.: НТУУ «КПІ», 2001. — № 4.

8. Hansen F. and Oleshchuk V.A.: Conformance Checking of RBAC Policy and its Implementation, The First Information Security Practice and Experience Conference, ISPEC 2005, Singapore, LNCS, Volume 3439, pp. 144—155, 2005.

9. Ленков, С.В. Методы и средства защиты информации. В 2-х томах / Ленков С.В., Перегудов - Д.А., Хорошко В.А., Под ред. В.А. Хорошко. - К. :

Арий, 2008.

10. Проблемы управления информационной безопасностью: Сборник трудов Института системного анализа Российской академии наук / Под ред. д-ра техн. наук, проф. Д. С. Череіпкина. — М.: Едиториал УРСС, 2002. — 224 с.

11. Что такое Firewall? [Електронний ресурс] – 2019. – Режим доступу до ресурсу: https://2ip.ru/article/firewall/.

12. Стаття 114. Шпигунство. [Електронний ресурс] – 2019. – Режим доступу до ресурсу: http://radnuk.info/komentar/kruminal/osobluva/287-rozd1/4401—114-.html.

13. Шлапаченко В. М. Кримінальна відповідальність за шпигунство: монографія / В. М. Шлапаченко. — К. : Центр навч., наук. та період. видань НА СБ України, 2015. — 348 с.

14. Кібербезпека чи Інформаційна безпека? | Блоги | Компьютерное Обозрение. ko.com.ua (ukr). Процитовано 2019-06-03.

15. Герберт Шилдт. Полный справочник по C# = C#: The Complete Reference. — М.: Издательский дом «Вильямс», 2004. — С. 26—27. — 752 с.

16. А. Хейлсберг, М. Торгерсен, С. Вилтамут, П. Голд. Язык программирования C#. Классика Computers Science. 4-е издание = C# Programming Language (Covering C# 4.0), 4th Ed. — СПб.: «Питер», 2012. — 784 с.